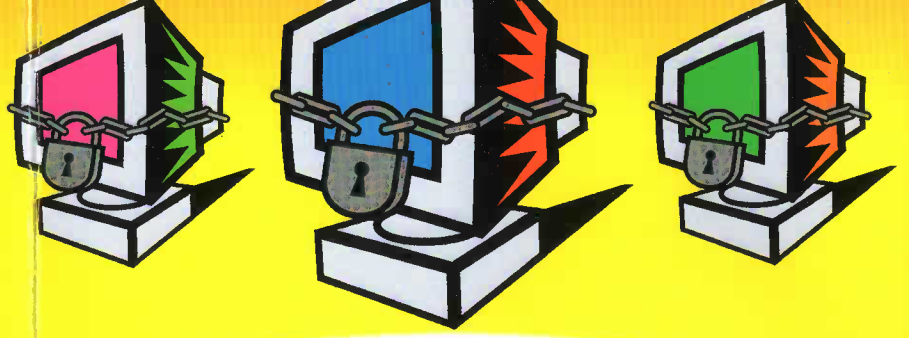


QUÉ ES LA SEGURIDAD INFORMÁTICA

HUGO D. SCOLNIK

PAIDÓS

Otros títulos de la colección

Daniel Cardinali
Qué es el sueño

Fernando Juan Pitossi
Qué son las células madre

Fidel A. Schaposnik
Qué es la física cuántica

Gabriel Rabinovich y Jorge Geffner
Qué es el sistema inmune

Hugo D. Scolnik

Qué es la seguridad informática

Prólogo
Nora Bär

PAIDÓS

Diseño de cubierta: Gustavo Macri
Imagen de cubierta: Gustavo Macri

Directora de colección: Nora Bär

Scolnik, Hugo D.
Qué es la seguridad informática -1ª ed.- Ciudad Autónoma de Buenos Aires: Paidós, 2014.
176 pp.; 19x12 cm.

ISBN 978-950-12-0169-7

1. Informática.
CDD 005.3

1ª edición en Argentina, noviembre de 2014

Reservados todos los derechos. Queda rigurosamente prohibida, sin la autorización escrita de los titulares del *copyright*, bajo las sanciones establecidas en las leyes, la reproducción parcial o total de esta obra por cualquier medio o procedimiento, incluidos la reprografía y el tratamiento informático.

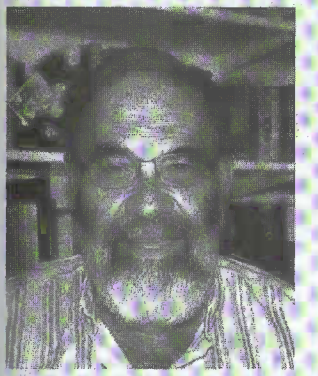
© 2014, Hugo D. Scolnik
© 2014, de todas las ediciones:
Editorial Paidós SAICF
Publicado bajo su sello PAIDÓS®
Independencia 1682/1686,
Buenos Aires – Argentina
E-mail: difusion@areapaidos.com.ar
www.paidosargentina.com.ar

Queda hecho el depósito que previene la Ley 11.723
Impreso en la Argentina – *Printed in Argentina*

Impreso en Master Graf,
Moreno 4794 - Munro,
Provincia de Buenos Aires,
en octubre de 2014.

Tirada: 3.000 ejemplares
ISBN 978-950-12-0169-7

Acerca del autor
Hugo D. Scolnik



Hugo D. Scolnik se graduó de licenciado en Matemática en la Universidad de Buenos Aires (UBA) y obtuvo su doctorado en la Universidad de Zúrich, Suiza. También es doctor honoris causa por la Universidad de Cuyo.

Recibió el Premio Konex de Platino 2003 en Ciencia y Tecnología, el Premio Manuel Sadosky en Informática otorgado por la Cámara de Empresas del Software y Servicios Informáticos en 2005, el Premio Segurinfo a la mejor trayectoria en seguridad informática en 2006 y el Premio Segurinfo al mejor producto de seguridad informática 2008.

En 2011, se le otorgó el título de Gran Maestro de la UBA.

Publicó trabajos en revistas internacionales sobre teoría de autómatas, inteligencia artificial, modelos matemáticos, matemática aplicada y criptografía.

Dirigió numerosas tesis de computación y de matemática en distintas universidades. Es *referee* y miembro del comité editorial de varias revistas internacionales. Trabajó como consultor nacional e internacional en seguridad informática, computación, modelos matemáticos e inteligencia artificial.

Dictó conferencias en facultades e institutos de la UBA, Alicante, Campinas, Cambridge, Chile, Columbia, Córdoba, Nueva York, Cuyo, Minnesota, Porto Alegre, Río de Janeiro, San Pablo, Stanford, Southampton, Sussex, Tokio, Tucumán, en el University College de Londres, en el Massachusetts Institute of Technology (MIT), en la Universidad Rice, en la Autónoma de México, en la de Zúrich, en la Politécnica de Cataluña y la Politécnica de Madrid. También en el Instituto de Investigación Operativa de la Universidad de Zúrich, en el Instituto Tecnológico Federal Suizo (ETH), en la Unesco de París, en la United Nations Industrial Development Organization (UNIDO) de Viena, en el Instituto Nacional de Planificación de El Cairo, en el International Institute for Applied System Analysis de Luxemburgo y en las Naciones Unidas, Nueva York.

Actualmente es profesor consulto titular del Departamento de Computación de la Facultad de Ciencias Exactas y Naturales de la UBA, y profesor y subdirector del Posgrado en Seguridad Informática de la misma universidad.

Índice

Prólogo, *Nora Bär*. 11
Agradecimientos 15
Introducción 17

Capítulo 1.
¿Cuáles son los enemigos que acechan nuestro universo digital? 23
Donde se siguen las huellas de virus, troyanos, gusanos, *cookies*... y del secuestro de páginas web

Capítulo 2.
¿Qué es la criptografía?. 39
Donde se relata una breve historia de la encriptación de mensajes, de las máquinas Enigma y Lorenz, y de lo que sucedió en la Segunda Guerra Mundial

Capítulo 3.
¿Cómo llega la encriptación al mundo civil?. 97
Acerca de Lucifer, de la televisión codificada y de cómo administrar claves cada vez más numerosas

Capítulo 4.
¿Qué es la criptografía de clave pública? 119
Acerca de rompecabezas, platos rotos, números primos y una pizca de azar

Capítulo 5.
¿Existen los sitios seguros? 135
Acerca de delincuentes del mundo virtual, candados, puertas traseras y técnicas de encriptación que se consideran armas

Capítulo 6.
¿Hacia una seguridad inquebrantable? 157
Acerca de la computación y la criptografía cuántica

Capítulo 7.
¿Cómo amenazan nuestra privacidad? 163
Acerca de la seguridad, privacidad, confiabilidad, los delitos informáticos y la información pública

Bibliografía 171

Prólogo

Hace un par de años, la publicación mundial de documentos secretos de la diplomacia norteamericana por la organización Wikileaks causó una conmoción y convirtió a Julian Assange en un héroe controvertido e inesperado. Aunque había sido tema de filmes de Hollywood y novelas de suspenso, de la noche a la mañana la guerra en gran escala que se libra en el universo digital se volvió evidente para el gran público.

Prácticamente ya no pasa un día sin que nos desayunemos con noticias sobre esta singular batalla global que abarca tanto a gobiernos, como a empresas o a simples individuos. En junio de 2013 los titulares dieron a conocer con grandes letras el caso Edward Snowden, que hizo públicos, a través de los diarios *The Guardian* y *The Washington Post*, documentos clasificados como *Top Secret* sobre programas de vigilancia masiva de la Agencia Nacional de Seguridad de los Estados Unidos.

En su artículo del 21 de julio para *Time*, "Guerra Mundial Cero", Lev Grossman menciona que en el último año fueron

hackeadas en los Estados Unidos 3000 compañías. Filtraciones de fotos y videos íntimos de celebridades, tomados de sitios de almacenamiento *online* supuestamente seguros escandalizan y ponen de relieve la vulnerabilidad de los usuarios del paraíso informático que construimos y al que estamos transfiriendo nuestra vida en forma de datos.

Piratas de distinta habilidad pueden conocer nuestros recorridos, "el colegio al que asisten nuestros hijos, los horarios de la familia y hasta lo que comimos el domingo, casi toda nuestra vida está hoy registrada en uno o más dispositivos digitales que pasan la mayor parte del tiempo en línea de forma autónoma", escribió Ariel Torres en el diario porteño *La Nación* a propósito de uno de estos episodios.

En este escenario digital, los errores de software se comercializan por cientos de miles de dólares, pero la seguridad informática parece no tener precio. Cómo proteger datos personales está pasando a ser un tema que debería interesar a todo aquel que tiene una cuenta bancaria o, simplemente, un celular.

No es sencillo de resolver. La problemática combina las vulnerabilidades propias de programas cada vez más complejos, la "viralización" de los datos digitales por las redes sociales y la propia naturaleza de Internet, y las conductas humanas que, por lo menos por ahora, todavía no llegan a adaptarse a este mundo interconectado en el que la información fluye libremente... incluso a lugares a donde no quisiéramos que llegara.

Como escribe el autor de este libro revelador e imprescindible, la capacidad de generar cada dos días más información digital que en toda la historia de la humanidad presenta enormes desafíos a la privacidad y la protección de datos

personales. Hugo Scolnik, matemático, pionero de la computación y de Internet, y un referente indiscutido en este tema en el país y en la región, bucea en busca de sus orígenes y los plantea en forma accesible al lector no experto.

A lo largo de sus páginas podemos aprender qué son los "troyanos" y los "cortafuegos", cómo administrar contraseñas o qué es la encriptación de mensajes. Estos términos y muchos otros, que a la mayoría nos resultan abstrusos, hoy están empezando a formar parte del léxico habitual de todo habitante del siglo XXI. Scolnik nos permite entrever las bambalinas de esta novela de espías y piratas en la que estamos inmersos aun sin quererlo. Pero también, y esto es tal vez más importante, nos ofrece medidas prácticas para protegernos.

Nora Bär,
agosto de 2014

Agradecimientos

A María Graciela y a mis hijos Daniel, Marina y Fernando.

A mi colega, amigo y socio criptógrafo, Juan Pedro Hecht, por tantos años de trabajo en común.

A los profesores Alberto Dams, Hugo Pagola, Graciela Pataro, Ricardo Rivas y Raúl Saroka, por haber compartido esa aventura exitosa que fue la creación del Posgrado en Seguridad Informática de la UBA.

A Néstor Masnatta, por haberse encargado durante tantos años con gran solvencia de las prácticas de mis cursos de Criptografía en la Facultad de Ciencias Exactas y Naturales de la UBA.

A mis colegas españoles Fernando Acero, Jorge Ramió Aguirre y Luis Hernández Encinas, por haberme invitado a participar de sus congresos y darme su apoyo en interminables investigaciones.

A mis amigos y socios de Certisur SA.

A Stella Maris Spurio, por su excelente apoyo técnico en el procesamiento de diversas imágenes.

A mi amigo Sergio Orce, por sus sugerencias. Y, finalmente, a la editora de esta colección, Nora Bär, sin cuyo entusiasmo y persistencia este libro jamás se habría completado.

Introducción

Las organizaciones gastan millones de dólares en cortafuegos y dispositivos de seguridad, pero tiran el dinero porque ninguna de estas medidas cubre el eslabón más débil de la cadena de seguridad: la gente que usa y administra las computadoras.

KEVIN MITNICK

Se vaticina que para el final de esta década existirán unos 22.000.000.000 de dispositivos conectados a Internet. En 2010 el CEO de Google, Eric Schmidt, afirmó: "Desde el amanecer de la civilización hasta el 2003, se crearon más de 5 Exabytes (equivale a 5 x 10^{18} caracteres) de información. En la actualidad, esta cantidad se está generando cada dos días".

La producción de datos no solo aumenta; explota. El crecimiento exponencial es tan grande que el 90% de los datos guardados en la actualidad han sido creados en los dos últimos años. Estos fenómenos son una consecuencia natural del desarrollo tecnológico, que implica grandes avances en

el acceso a la información, mejoras en el sistema productivo y la educación, etc., pero al mismo tiempo, presenta enormes desafíos a la privacidad y a la protección de datos personales, y múltiples posibilidades de delitos informáticos.

Es precisamente de estos temas que se ocupa la seguridad informática pero, antes de avanzar, vale la pena hacer una reflexión sobre el uso de las tecnologías de la información y las comunicaciones (TICs) por parte de las personas que no poseen conocimientos especializados y que, a menudo, no tienen ningún interés en ellos. Las TICs deberían ser como los servicios normales de electricidad, teléfono o agua corriente. O sea que cuando una persona se conecta a Internet [*World Wide Web* o, simplemente, la Web] habría que asegurarle la prestación sin mayores requisitos. Eso lamentablemente no sucede; es la gran deuda que las TICs tienen con la gente que las usa cada vez más para todo tipo de actividades. En otras palabras, el ciudadano está obligado a adquirir una cantidad de conocimientos básicos para poder sobrevivir en la jungla de las PC [*Personal Computer*; computadoras personales], *notebooks*, teléfonos inteligentes [*smartphones*] y tabletas que se utilizan para realizar operaciones bancarias, obtener datos de salud, redactar todo tipo de documentos y planillas de cálculo, comunicarse con amigos, parientes, enemigos e innumerables actividades más cuya variedad y complejidad crecen continuamente.

Una solución posible es la adopción de la computación en la nube [*cloud computing*], donde un usuario se conecta por Internet a servidores remotos que disponen de todos los programas que necesita, y para almacenar allí sus datos. Este modelo tiene diversas ventajas pues todos los problemas de seguridad y resguardo de la información recaen en el proveedor del servicio, además de que se dispone en todo momento de las últimas versiones de software. Otra ventaja es que el usuario puede pagar por el uso efectivo, o sea que no está obligado a comprar un software que quizás solo use en forma ocasional. Una objeción frecuente a la computación en la nube se refiere a la privacidad de la información, algo que se resuelve con un diseño adecuado de encriptación de los datos.

La seguridad informática puede definirse como el conjunto de procesos destinados a proteger la disponibilidad, la privacidad y la integridad de los datos, sea de personas o instituciones. Para lograr estos objetivos es imprescindible utilizar técnicas varias, pero entendiendo que no existen sistemas infalibles que garanticen la seguridad total.

A fin de tener una apreciación de la dimensión del problema es recomendable entrar al sitio <map.ipviking.com>, donde podrán ver en tiempo real, es decir, a medida que se detectan, los ataques cibernéticos entre países (los que se muestran gráficamente son solo una parte de los reales).

La seguridad informática es una especie de batalla continua entre atacantes y defensores que les exige a los profesionales del área estar permanentemente informados, dado que aparecen novedades en forma cotidiana.

Una pregunta usual es: ¿por qué existen debilidades en los sistemas informáticos que permiten que sean atacados? La respuesta es que normalmente hay errores en los programas [*bugs*] en el nivel del sistema operativo (Windows, Linux, Android...), de los navegadores (Internet Explorer, Google Chrome, Mozilla, Firefox...) y de los programas de aplicaciones de toda clase. ¿Y no se pueden prevenir o detectar los errores? Lamentablemente, las técnicas formales de verificación de programas que permiten asegurar que un

software determinado solo hace lo que debe hacer se limitan por el momento a sistemas "pequeños", o sea que no son aplicables hasta ahora en forma eficaz a productos de gran complejidad. Por ese motivo los errores van surgiendo en forma experimental, o por el análisis detallado de aspectos parciales de los sistemas. Esto conduce a la generación de parches continuos, nuevas versiones, etc., que intentan remediar lo que se va descubriendo.

En su origen la seguridad informática se restringía fundamentalmente a la seguridad, sobre todo física, de los equipos. Si consideramos que las primeras computadoras estaban asociadas con el esfuerzo bélico, resulta natural que protegerlas fuera un objetivo primordial. Hoy en día, la seguridad física es sin duda muy importante, tanto desde el punto de vista de los dispositivos personales como de los grandes centros de datos, pero ese es un concepto obvio y evidente para cualquier persona. Lo más complejo de entender e internalizar se refiere a la seguridad "no física". En ese sentido existen múltiples problemas cuando usamos PC, *notebooks*, teléfonos inteligentes y tabletas.

Para operar con seguridad nuestros dispositivos digitales, tenemos que poder responder, por ejemplo, a estas preguntas:

- ¿Qué son los virus, *malware*, troyanos, gusanos, *cookies*... y el secuestro de páginas web?
- ¿Qué son los cortafuegos [firewalls]?
- ¿Cuáles son las debilidades del sistema operativo (Windows, Linux, Android)?
- ¿Qué son las intrusiones y los problemas de autenticación?

- ¿Qué es la encriptación de mensajes?
- ¿Se pueden quebrar códigos usando la "fuerza bruta" (es decir, probando todas las combinaciones posibles)?
- ¿Cuáles son los métodos simétricos modernos?
- ¿Estos métodos son seguros?
- ¿Cómo evaluar un método simétrico?
- ¿Qué es el *hashing* (digestos de mensajes)?
- ¿Cómo administrar contraseñas [*passwords*]?
- ¿Cómo almacenar claves de usuarios en una computadora?
- ¿Qué es una autenticación remota?
- ¿Cómo se encriptan las claves públicas?
- ¿Qué es la firma digital?
- ¿Qué es HTTP y HTTPS? ¿Son sitios seguros?
- ¿Qué significa cuando vemos una barra verde en un sitio de Internet?
- ¿Qué es la esteganografía (ocultar mensajes dentro de otros mensajes)?
- ¿Qué peligros presenta el uso de Wi-Fi (conexiones inalámbricas) y celulares?
- ¿Cómo utilizar el correo electrónico [*e-mail*] en forma segura?
- ¿Qué es el *phishing*?
- ¿Qué es la criptografía cuántica?
- ¿Qué son las computadoras cuánticas?
- ¿Qué técnicas de cibertecnología amenazan la privacidad?
- ¿Es defendible el uso de *cookies*?
- ¿Qué problemas surgen con la información pública?

Esta será, entonces, nuestra "hoja de ruta".

¿Cuáles son los enemigos que acechan nuestro universo digital?

Donde se siguen las huellas de virus, troyanos, gusanos, *cookies*... y del secuestro de páginas web

En general hablamos de *malware* o código malicioso cuando nos referimos a aquel cuyo objetivo es lograr que nuestro dispositivo realice tareas definidas por esos programas y no por el usuario, que pueden variar desde acciones "inocentes", como distorsionar las imágenes en una pantalla, hasta la destrucción total del software y los datos almacenados mediante el borrado de los discos.

Los virus se replican dentro de los equipos y se transmiten por vías diversas –por ejemplo, los *e-mails*– con el objetivo de propagarse a otros dispositivos.

Los troyanos son una clase de virus generalmente disfrazados dentro de programas "deseables", como juegos, o en los archivos adjuntos a correos que despiertan el interés del destinatario. Un caso habitual es que uno busca programas que tienen un costo por los canales normales de comercialización, pero aparecen sitios en la Web que los ofrecen gratuitamente. En la inmensa mayoría de los casos esos regalos tentadores son justamente "caballos de Troya" que infectan el equipo en el que se instalan.

Los gusanos [*worms*] son también una clase de virus que se propaga en las redes replicándose en forma continua. Su objetivo es multiplicarse de manera incontrolada, sin intervención humana, y generalmente consumen los recursos del sistema hasta llegar a paralizarlo.

El primer gusano data de 1988 y se llamó "Morris" por su creador, Robert T. Morris, quien lo diseñó cuando era estudiante en la Universidad de Cornell –actualmente es profesor del MIT–, para "medir" la capacidad de Internet (en realidad la de su predecesora, la red ARPANET). Sus efectos fueron dramáticos: en pocas horas quedaron fuera de servicio todos los centros importantes de los Estados Unidos, desde la red de Defensa hasta la NASA, a pesar de que ese gusano atacaba solo computadoras con sistemas Unix (versión BSD Berkeley). En esa época eran muy populares los sistemas de Sun Microsystems y VAX de Digital que no fueron afectados. Si ese no hubiese sido el caso, los efectos podrían haber llegado a niveles de catástrofe.

Los expertos analizaron rápidamente el problema hasta ubicar el código que lo causaba, y crearon una "vacuna" que lo solucionó. Esto nos lleva a la actualidad, pues la mejor manera de defenderse es instalar antivirus, o sea, programas que contienen "vacunas" cuya misión es controlar todo lo que llega a nuestro dispositivo. Por ejemplo, analizar un archivo adjunto a un *e-mail* antes de abrirlo.

Los antivirus deben actualizarse continuamente debido a la aparición de nuevas amenazas y pueden fallar por una razón obvia. Supongamos que el antivirus se actualizó a las 10 am y aparece un nuevo virus a las 10.01 am. Mientras la empresa proveedora modifica su antivirus para lidiar con la novedad, el flamante virus infecta nuestro dispositivo. Por

ello es muy recomendable revisar a fondo nuestros equipos en forma periódica (usamos el neologismo "escanear" por la palabra inglesa *scanning*), lo que permite encontrar código malicioso en programas muy antiguos.

Surge la pregunta de cuáles son los motivos que impulsan a crear un código malicioso. Hay respuestas diversas; por ejemplo, que se diseñan como métodos de ciberataques entre países o que sirven como "tarjeta de presentación" a los programadores que los implementan frente a las grandes empresas que eventualmente los contratan por sus habilidades técnicas. Ese fue el caso del virus Stuxnet, creado por Israel y los Estados Unidos para afectar el programa nuclear iraní, que atacó los sistemas de control industriales SCADA utilizados en muchas infraestructuras críticas en todo el mundo. En Wikipedia podemos leer:

> La empresa europea de seguridad digital Kaspersky Lab describía a Stuxnet en una nota de prensa como "un prototipo funcional y aterrador de un arma cibernética que conducirá a la creación de una nueva carrera armamentística mundial".

Lo que sí se observa es el comienzo de una declinación de los ataques clásicos de virus, *malware*, etc. De hecho, Conficker (en 2008) fue el "último" de los ataques masivos; y en cambio, existe un aumento de código malicioso para Mac, teléfonos inteligentes y dispositivos con Android.

¿Quiénes son los que se dedican a penetrar sistemas, escribir códigos maliciosos y otras actividades conexas? Los llamados *hackers*, cuyos máximos objetivos son introducirse en sistemas –cuanto más protegidos mejor, pues presentan un mayor desafío–, robar información y procurar no dejar

rastros detectables. Un *hacker* de alto nivel lo primero que hace cuando logra entrar en un sistema es instalar diversos mecanismos para poder volver a ingresar en el futuro, aun cuando su método original sea descubierto y removido.

Hace años los *hackers* eran pocos y debían tener muchos conocimientos técnicos. Hoy en día es posible obtener en la Web algunos programas semiautomáticos que analizan una gran cantidad de direcciones en Internet y detectan vulnerabilidades que pueden ser explotadas. En otras palabras, una persona no muy especializada puede dejar ejecutando un programa durante unas cuantas horas, lo que le reportará resultados para penetrar en equipos diversos. Por ese motivo la cantidad de ataques ha proliferado en forma impresionante. Por ejemplo, hay países donde, si uno se conecta a la Web vía Wi-Fi, es atacado en menos de un minuto.

Hemos utilizado la palabra "*hacker*" por lo popular del término, pero sin ninguna connotación peyorativa, pues no equivale a delincuente digital. Un *hacker* puede ser un investigador, un académico interesado en la seguridad informática, etc. En Wikipedia, leemos:

> Un *hacker* no es un delincuente, vándalo o pirata informático con altos conocimientos técnicos [a los que prefiere llamar *crackers*], sino [...] todo aquel que trabaja con gran pasión y entusiasmo por lo que hace. De ahí que el término "*hacker*" pueda y deba extrapolarse a otros ámbitos [...], por ejemplo, el científico.

Uno de los efectos más molestos del *malware* que afecta los equipos que usan Windows es la aparición de ventanas emergentes con propagandas no solicitadas (los *pop-ups*), el cambio de las páginas de inicio, la instalación de barras de

tareas y hasta el "secuestro" de los buscadores que uno eligió. Veamos este tema con más detalle dado que los programas maliciosos son cada vez más frecuentes. En principio, no existe una solución universal que nos proteja para siempre porque los atacantes evolucionan y continuamente aparecen nuevas técnicas que derrotan a los mecanismos de protección. Por ejemplo, hay muchos programas maliciosos que cambian todo el tiempo. Son los denominados "virus polimórficos", que intentan evadir la detección modificando su patrón de *bytes* con cada infección. Esto es así porque los antivirus buscan segmentos "reconocibles" de código para identificarlos.

Una clase de ataques mediante *malware* es la instalación no deseada en una computadora de los llamados *keyloggers* (registradores de teclas), que graban todo lo que escribe una persona en un archivo que luego será enviado al *hacker* por la Web. Obviamente el objetivo es capturar claves u otra información sensible. Por ese motivo hoy en día casi todos los bancos poseen la opción de un teclado virtual consistente en una imagen de un teclado normal al que se le puede mezclar la posición de los símbolos. Para escribir se usa el ratón, y eso previene tanto la existencia de *keyloggers* como del posible *hackeo* de una conexión mediante redes públicas no seguras.

Los ataques a las páginas de inicio y a los buscadores son muy frecuentes. A veces uno busca un producto determinado en la Web como reproductores de video, juegos, etc., y al final encuentra que aparecieron cambios inesperados en su PC, desde barras de marcadores hasta modificaciones de la configuración del buscador.

Las *cookies* son pequeños archivos que un sitio de Internet envía y almacena en el navegador del usuario para con-

seguir cierta información, como su historial de acceso a ese sitio, su idioma preferido, etc. En principio, puede guardar datos de identificación, como las claves de acceso, para no tener que reingresarlos con frecuencia, la información de la computadora y qué navegador utiliza. Obviamente esto presenta problemas de seguridad para el usuario, pues son un medio para que las campañas de marketing le hagan llegar publicidad. Esto es algo que uno advierte cuando luego de realizar una búsqueda comienzan a aparecer propagandas relacionadas con ese tema.

Una aplicación común de las *cookies* es en el comercio electrónico pues se usan para "recordar" los artículos incorporados a un carrito de compras.

Estos archivos pueden habilitarse, borrarse o bloquearse mediante la configuración del navegador. Por ejemplo, en el caso de Internet Explorer, hay que ir a Herramientas ➔ Opciones de Internet ➔ Privacidad, y allí se define el nivel de aceptación de las *cookies*.

En el caso de Google Chrome:

1. Marcar el icono del menú de Chrome .
2. Ir a Configuración.
3. Ir a Mostrar opciones avanzadas.
4. En la opción "Privacidad", marcar Configuración de contenido.
5. Opciones:
   - ◉ Permitir que se almacenen datos locales (recomendado).
   - ○ Conservar datos locales solo hasta que salgas del navegador.
   - ○ No permitir que se guarden datos de los sitios.
   - ☐ Bloquear los datos de sitios y las *cookies* de terceros.
6. Cliquear Ok para guardar las opciones elegidas.

Las *cookies* viajan entre la PC del usuario y el sitio web que visita transportando información. Si la conexión se realiza en forma no encriptada (véase la próxima sección), tal como sucede en una red Wi-Fi abierta, un *hacker* puede interceptarla y usarla para sus fines. Un sitio web puede obligar a que un navegador utilice o no comunicaciones encriptadas. En el caso de que no lo haga, las *cookies* pueden ser explotadas por los *hackers*. Por ese motivo nunca hay que poner datos personales en sitios inseguros.

El problema es que, para funcionar, muchos sitios (como YouTube, Gmail y Yahoo) requieren que las *cookies* estén habilitadas y por eso es conveniente eliminarlas con frecuencia para evitar los problemas de seguridad. Esto puede hacerse con distintos productos, por ejemplo, con CCleaner, y en Google Chrome, con la opción Borrar datos de navegación, etc.

Para usuarios de Linux que utilizan navegadores como Firefox u Opera la recomendación es entrar a los foros especializados para ver cómo se almacenan las *cookies* y cómo eliminarlas.

Las acciones recomendadas son las siguientes:

- Instalar un antivirus como Avast, AVG, NOD32, Panda, Kaspersky, etc. (hay versiones para Windows, Linux, Mac, Android, IOS).
- Instalar un protector contra *malware* como Malwarebytes, Iobit Malware Fighter, Adguard, Winzip Malware Protector, CCleaner, etc.

Algunos de estos productos tienen mecanismos de protección de la página principal, o sea que si uno define que al entrar a Internet aparezca un determinado sitio, en teoría no puede ser cambiado. Estos productos requieren actualizar continuamente sus bases de datos, algo que puede hacerse de manera automática o manual. Si el escaneo no resuelve los problemas, se puede probar con Microsoft Safety Scanner o el Spybot Search & Destroy. Y si tampoco se tiene éxito con lo anterior, hay que apagar la PC y volver a encenderla apretando la tecla F8 (si se usa Windows) hasta que aparezca una ventana que ofrezca arrancar en modo seguro. Hágalo y luego proceda a realizar un *full scan* (escaneo completo) con el *antimalware* que haya elegido.

- Si nada de lo anterior termina con las modificaciones en su buscador (Internet Explorer, Google Chrome, Yahoo, etc.), hay que desinstalarlo completamente y volverlo a instalar recurriendo a la página oficial del proveedor para tener la última versión que, seguramente, tendrá más medidas de seguridad.[1]

Los problemas mencionados se agudizan por las malas prácticas comerciales de muchas empresas. Por ejemplo, uno desea descargar un producto y aparecen botones confusos que terminan instalando programas no deseados. O, en el medio de la instalación, le piden definir un buscador y las

1. Véase <www.microsoft.com/es-es/security/resources/hijacking-whatis.aspx>. Para los usuarios de Internet Explorer, véase <www.microsoft.com/es-es/security/pc-security/browser-hijacking.aspx>. Un sitio recomendable para productos gratuitos es <descargar.cnet.com>.

opciones no incluyen el que uno prefiere, obviamente con el propósito de forzarnos a usar los que ellos quieren. Otra técnica es pronosticar males incurables en su equipo si no compra un producto determinado. Es común que un servicio remoto de antivirus "detecte" muchos virus que solamente se pueden remover si uno compra la versión completa que están ofreciendo.

¿QUÉ SON LOS CORTAFUEGOS [FIREWALLS]?

Un cortafuegos es un software o hardware cuyo objetivo es bloquear los accesos no autorizados a nuestra computadora o a nuestra red local según criterios que el usuario define.

Los usuarios individuales suelen utilizar los implementados por software. Existen muchos gratuitos, como Zone Alarm, Comodo, PSafe, etc., para Windows, y ClearOS, IPCop, Monowall, etc., para Linux. Generalmente cada sistema operativo y/o distribuciones particulares, como Ubuntu en Linux, tienen un cortafuegos propio, aunque no siempre son los más recomendables. Al instalar un cortafuegos este comienza a pedir autorización para ejecutar cada programa y tiene una opción que permite recordar la decisión del usuario para, en teoría, no volver a preguntar lo mismo en la siguiente ejecución.

De todos modos, es necesario puntualizar que, aunque es imprescindible instalar cortafuegos, estos no constituyen una garantía absoluta de la seguridad informática.

¿CUÁLES SON LAS DEBILIDADES DEL SISTEMA OPERATIVO (WINDOWS, LINUX, ANDROID...)?

Los sistemas operativos están constituidos por un conjunto de programas destinados a gestionar los recursos de hardware de los dispositivos y proveer los servicios necesarios para las aplicaciones. Su ejecución tiene prioridad sobre cualquier otro programa. Constan de una parte central, denominada núcleo [*kernel*], cuya misión esencial es la de permitir a los programas que se ejecuten con un acceso seguro al hardware y gestionando los recursos que se necesiten. Lo importante es que le ahorran al programador tener que lidiar con problemas muy complejos que requieren un alto grado de información a nivel electrónico.

También existen programas como los controladores [*drivers*], que son los que permiten manejar los periféricos –pantallas, ratones, impresoras, etc.–, que trabajan de modo conjunto con el sistema operativo, formen o no parte de él.

La historia del desarrollo de los sistemas operativos es importante para entender algunos de los problemas de seguridad existentes en la actualidad, y por eso la veremos sucintamente.

Los primeros sistemas operativos tenían que lidiar con el procesamiento en lotes [*batch*], esto es, se ejecutaba un programa y, al terminar de hacerlo, se continuaba con el siguiente en forma secuencial. Luego los problemas se complejizaron con, por ejemplo, los sistemas multiusuario, en los cuales se procesan diversas tareas en forma concurrente, pues hay que administrar los recursos (memoria, discos) de forma tal que no existan errores en las ejecuciones. Y como era esperable, a mayor complejidad se incrementa la probabilidad de que se produzcan errores de diversa clase que eventualmente pueden ser aprovechados por los *hackers*.

En el contexto de este libro no nos interesa profundizar más sobre las clases de sistemas operativos ni en otros detalles técnicos, tema acerca del cual existen muy buenos textos. Pero sí queremos concentrarnos en algunas preguntas relevantes para la seguridad. La primera es: ¿hay un sistema operativo superior a todos los otros?

La respuesta es negativa, pues cada uno tiene sus ventajas y desventajas y, además, existen muchos mitos cuasirreligiosos en estos temas. Veamos algunas de las características de los más populares.

WINDOWS

Las computadoras personales originales utilizaban un sistema operativo bastante elemental llamado MS-DOS [Microsoft Disk Operating System; sistema operativo de disco de Microsoft] parecido a otro lenguaje que se llamaba CP/M y que se manejaba con comandos tipo Unix. Por ejemplo, *cd c:\documentos* servía para "moverse" desde la raíz del disco *C* al directorio *documentos* (*cd* significa *change directory*, o sea, cambiar de directorio). Una vez en él, un comando como *dir* listaba todo su contenido. Su origen se remonta a 1980, cuando IBM lanza la primera PC y le acerca a Microsoft (empresa fundada en 1975) un proyecto de sistema operativo llamado Chess (ajedrez), indispensable para comunicarse con la máquina. Pero Microsoft se concentra en el desarrollo del MS-DOS. En ese momento IBM creía que el negocio era la venta de computadoras, mien-

tras que Microsoft –como su nombre lo indica– apostó al software.

En 1985 ya existía un creciente interés por las interfaces gráficas de usuario [*Graphical User Interface*; GUI] debido a las creaciones pioneras de un equipo del Stanford Research Institute liderado por Douglas Engelbart que originalmente trabajaba en modo texto y luego evolucionó a un sistema gráfico creado por los investigadores del Xerox Parc en Palo Alto, California.

Desde este laboratorio se han gestado ideas tecnológicas de gran relevancia para la industria, como la impresión láser o la programación computacional, y otras de gran impacto antes de la era post-PC (Pérez, 2013), como el protocolo Ethernet, el WYSIWYG (*What You See Is What You Get*, es decir, lo que ve en la pantalla es lo que saldrá por impresora), o alguno de los estándares de comunicación a través de Internet. Estos logros fueron fruto de la visión emprendedora de la compañía y de su intención de establecer un laboratorio de investigación de referencia internacional a finales de la década de 1960. Para ello, Jack Goldman, uno de los principales investigadores de Xerox, consiguió que el responsable financiero de la Universidad de Washington, un físico llamado George Pake que se especializaba en resonancia magnética nuclear, financiase en 1969 un segundo centro de investigación para Xerox.

Pero pasemos a la historia de Apple y Xerox. En 1979 Apple comenzó un programa de venta de acciones a grandes grupos inversores como forma de financiación, y uno de los que contactó fue Xerox. El Apple II estaba en todo su apogeo, en especial tras la salida al mercado de Visicalc (una planilla de cálculo que atrajo a una enorme cantidad de usuarios

pues permitía hacer muchas aplicaciones, como contabilidad, etc.) y algunos paquetes más de software que justificaban que las empresas adquirieran ordenadores personales para sus empleados. Por otra parte, el mercado educacional acababa de explotar y el Apple II era la computadora estrella con campañas especiales para los centros educacionales.

En Xerox Parc habían desarrollado en 1973 la primera PC, denominada Xerox Alto (dos años antes de que se fundara Apple), la cual a su vez evolucionó al modelo Xerox Star 8010 en 1981. Se trató del primer sistema comercial que utilizaba ventanas, íconos, ratón, *e-mail*, etc. Poco antes, en 1979, un equipo de la empresa Apple visitó Xerox Parc y procedió a diseñar una interface gráfica para un modelo que se denominó Lisa. Ese sistema de ventanas fue posteriormente utilizado por varios modelos de Apple, de los cuales Macintosh fue el más relevante. De hecho su nivel de resolución en blanco y negro era muy notable, considerando que salió al mercado en 1984... ¡Y que pesaba mucho!

Bill Atkinson, Andy Hertzfeld, Steve Capps y Larry Kenyon fueron los principales desarrolladores del software para Macintosh.

Sin duda estos trabajos pioneros inspiraron a Microsoft para trabajar en el desarrollo de lo que primero se denominó Interface Manager, aunque finalmente predominó el nombre Windows. El 20 de noviembre de 1985, Microsoft comenzó a vender la primera versión del nuevo sistema operativo, el Windows 1.0. La ventaja para el usuario final era que, en vez de tener que escribir los crípticos comandos de MS-DOS, bastaba con utilizar un ratón para hacer clic en las diferentes ventanas. También el hecho de que se pudieran tener distintas ventanas abiertas con programas diferentes, y

pasar de uno a otro con un simple clic (vale la pena observar que sin problema se pueden tener varias ventanas corriendo sendos programas en MS-DOS), era toda una novedad para la época.

¿QUÉ SON LAS INTRUSIONES Y LOS PROBLEMAS DE AUTENTICACIÓN?

Las intrusiones a un dispositivo como una PC pueden compararse con las que se dan en nuestras casas. Si cerramos todos los accesos, en principio nadie puede entrar. Pero en cuanto abrimos puertas o ventanas, hay que tomar precauciones.

Los ingresos no autorizados a nuestras computadoras se originan fundamentalmente por la conexión a una red como Internet. Toda computadora posee interfaces físicas (o sea, en su hardware) o lógicas (definidas por software) que se denominan "puertos". Ellas permiten la transferencia de datos de todo tipo, como el puerto de la impresora que la conecta a la computadora para poder enviarle la información correspondiente a un archivo que queremos imprimir. Análogamente hay puertos para los discos, el ratón, el teclado, los dispositivos de audio, las lectograbadoras de CD o DVD, los USB, la HDMI, etc. Cuando conectamos nuestra computadora a una red, la PC comienza a ser una parte integral de esta y, por lo tanto, necesita comunicarse con los dispositivos que la forman. En el caso de Internet, al conectarnos se nos asigna un número identificatorio llamado "dirección IP" [*Internet Protocol*]. Originalmente estas direcciones utilizaban la versión IPv4 (cuarta versión del protocolo) y consistían en cuatro blo-

ques de cuatro dígitos como máximo. Esto permitía disponer de 4.294.967.296 direcciones posibles. A principios de 2010, quedaban menos del 10% de IP sin asignar. En la semana del 3 de febrero de 2011, la Agencia Internacional de Asignación de Números de Internet [Internet Assigned Numbers Authority; IANA] entregó el último bloque de direcciones disponibles (33 millones) a la organización encargada de asignar IP en Asia, y el 10 de junio de 2014 se llegó al agotamiento total de las existentes. Si consideramos que, como dijimos al inicio, se espera que para fines de la presente década haya unos 22.000.000.000 de dispositivos conectados a la Web, está claro que era imposible continuar con el IPv4. Cabe destacar que también existe un desperdicio considerable de direcciones IP debido a que muchas de ellas se asignan a redes locales (LAN) que necesitan definir subredes a partir de la IP que se les asigna, y como todo se maneja con potencias de 2, bien puede suceder que la cantidad de subredes difiera bastante de una potencia de 2; eso conduce a tener "subdirecciones" no utilizadas. Por estos motivos se creó el protocolo IPv6, que permite disponer de 340.282.366.920.938.463.463.374.607.431.768.211.456 (2^{128} direcciones) –cerca de $6{,}7 \times 10^{17}$ (670.000 billones) de direcciones por cada milímetro cuadrado de la superficie de La Tierra–.[2] Esta cantidad gigantesca ha permitido llegar hasta el punto de reservar bloques de direcciones para las conexiones interplanetarias. Una vez que disponemos de una dirección IP podemos conectarnos a través de los puertos (en una PC con Windows hay unos 60.000). Por ejemplo, el puerto 25 se usa para el protocolo SMTP [*Simple*

2. Véase "IPv6", en <www.es.wikipedia.org>.

Mail Transfer Protocol] utilizado, como su nombre lo indica, para el correo electrónico.

Algo usual para transferir archivos es el protocolo FTP [*File Transfer Protocol*] que utiliza el puerto 21 que, si está mal configurado o desprotegido, es una puerta abierta para los troyanos.

Todo equipo debe estar protegido contra las intrusiones. Por ese motivo se crearon diversos mecanismos para asegurarse de que solo los usuarios autorizados puedan acceder a los archivos, la memoria, la CPU y otros recursos. Por lo general, en toda PC tenemos un usuario con privilegios, el administrador, y otros "normales" a los que aquel les concede derechos de acceso. Por ejemplo, el administrador puede definir directorios o archivos que solo sean de lectura, o sea, que no pueden ser modificados por otro usuario. En general, es muy recomendable dividir los roles aunque el equipo lo use solo una persona; es decir, que haya una cuenta de administrador para las tareas que requieren ese rol y otra sin tantas prerrogativas para el uso normal. Esto es útil porque si un *hacker* entra al sistema y toma posesión de la cuenta del administrador, puede hacer lo que quiera dentro de la PC.

¿Qué es la criptografía?

Donde se relata una breve historia de la encriptación de mensajes, de las máquinas Enigma y Lorenz, y de lo que sucedió en la Segunda Guerra Mundial

La idea básica de los métodos de encriptación consiste en transformar un mensaje de modo tal que sea inentendible salvo para los que posean la clave que permite descifrarlo. Los métodos se clasifican en simétricos, en los cuales se utiliza la misma clave para encriptar y desencriptar, y asimétricos, donde cada entidad (usuario, organismo) posee una clave privada y una pública.

La criptografía estudia, entre otros temas, los métodos de encriptación, y el criptoanálisis las técnicas para intentar quebrarlos. La criptografía consiste de un conjunto de técnicas al servicio de la seguridad informática y no debe confundirse con ella.

Los objetivos de la criptografía son generar mensajes que puedan asegurar:

- Confidencialidad: queremos enviar mensajes de modo tal que solamente sean legibles para el destinatario. Esto se consigue mediante el uso de claves.

- Integridad: cuando enviamos un mensaje queremos estar seguros de que este no haya sido alterado antes de llegar al destinatario. Esto se logra mediante los llamados *hashings*.
- Autenticidad: cuando recibimos un mensaje queremos verificar la identidad del emisor.
- No rechazo: cuando alguien envía un mensaje asumiendo un compromiso, por ejemplo, firmando un contrato en forma digital, queremos que no pueda ser repudiado.

Analizaremos estos conceptos con mayor detalle al tratar el tema de la firma digital.

La historia de la criptografía es apasionante pues estuvo motivada por razones militares, políticas o religiosas. En el Antiguo Egipto, hace unos cuatro mil quinientos años, el uso de los jeroglíficos ya constituía una barrera entre el común de la gente y los sacerdotes, pero a eso le agregaron algo ingenioso: se grababa en una piedra un mensaje extenso y simultáneamente se perforaban agujeros en otra piedra que se apoyaba sobre la primera. El mensaje real era el que se podía ver a través de las perforaciones, así que la copia de la piedra original y la de la piedra agujereada se podían hacer llegar al interlocutor por vías separadas. Esto podría considerarse como un primer ejemplo de esteganografía, o sea, el arte de ocultar mensajes dentro de otros mensajes.

El Atbash, un método usual entre los sabios hebreos utilizado unos cinco siglos antes de Cristo, consistía en un sistema de sustitución monoalfabético.

En el siglo V antes de Cristo los griegos crearon un dispositivo para cifrar mensajes, conocido como Escítala de

los lacedemonios o Escítala espartana. Estaba formado por un cilindro de madera en el cual se enrollaba una cinta de papiro, tela o cuero. Una vez enrollada, se escribía el mensaje longitudinalmente, de forma que en cada vuelta de cinta apareciese una letra por vez. Después se desenrollaba la cinta y se enviaba con un mensajero al receptor, quien contaba con un cilindro idéntico, por lo que podía volver a enrollar la cinta tal como se había hecho originalmente, recuperando así el mensaje. La cinta sola, sin el cilindro, era ilegible. El diámetro de la Escítala era la clave de este método de encriptación pues definía cada cuántos caracteres empezaba una vuelta.

Polibio fue un historiador griego nacido aproximadamente en el año 200 a. C., que inventó un método ingenioso para transmitir señales encriptadas. Su idea fue escribir el alfabeto en un cuadrado de 5 x 5, del siguiente modo:

	1	2	3	4	5
1	A	B	C	D	E
2	F	G	H	IJ	K
3	L	M	N	O	P
4	Q	R	S	T	U
5	V	W	X	Y	Z

Entonces, una letra se identificaba por sus coordenadas en la tabla, por ejemplo Q = 41, W = 52, etc. Polibio sugirió

que los mensajes se transmitiesen mediante antorchas: cuatro antorchas en la mano izquierda y una en la derecha significaban 41, o sea, la letra "q". Su idea fue muy apreciada por los criptógrafos contemporáneos pues tuvo la originalidad de transformar letras en números, de reducir el número de caracteres diferentes, como el caso IJ, y de dividir una unidad (letras) en partes separables (sus coordenadas).

Un avance importante tuvo lugar recién en el siglo XV cuando Leon Battista Alberti inventó los métodos polialfabéticos. Su principal limitación era que la persona que codificaba un mensaje debía indicar en el criptograma cómo elegir el siguiente alfabeto. Un hecho notable fue su invención del primer método de encriptación por "hardware" pues implementó la clave de sustitución polialfabética mediante dos discos concéntricos.

El siguiente adelanto se debió a Giovan Battista Bellaso, quien aparentemente fue el primero en proponer, en 1553, que las permutaciones se identificaran mediante códigos que debían intercambiarse entre las personas que querían comunicarse.

Por ejemplo, si decidían usar AB, eso indicaba que la sustitución debía ser esta:

A	B	C	D	E	F	G	H	I	L	M
N	O	P	Q	R	S	T	U	X	Y	Z

En cambio, sí aparecía por ejemplo VX, debía usarse otra permutación, etc.

Durante el Renacimiento proliferaron, principalmente en Italia, los métodos de encriptación.

Su popularidad condujo a que se crearan los secretarios de cifra, responsables tanto del cifrado de la correspondencia entre reyes, ministros y embajadores, como de criptoanalizar la correspondencia intervenida de otros Estados.

Por ejemplo, en España, el primer secretario de cifra conocido es Miguel Pérez de Almazán, nombrado por los Reyes Católicos. Pero fue Felipe II quien renovó las técnicas de cifrado y les dio un gran impulso, poniéndolas bajo la responsabilidad del secretario de cifra Luis Valle de la Cerda. Este estableció la Cifra General (para comunicarse con sus secretarios, sus embajadores y altos militares, y para que lo hicieran entre ellos) y la Cifra Particular, para las comunicaciones más confidenciales (Ribagorda Garnacho, 2010).

Recién en el siglo XIX se comenzó a desarrollar la criptografía científica, principalmente con los trabajos de Auguste Kerckhoffs, lingüista y criptógrafo holandés, publicados en una revista militar francesa. En 1883, Kerckhoffs estableció los siguientes principios para los sistemas criptográficos (en el lenguaje de la época):

- Deben ser inquebrables en la práctica, si no lo fuesen teóricamente.
- La difusión del método no debe afectar a los corresponsales.
- La clave debe ser memorizable y fácilmente cambiable.
- El cifrado debe ser telegrafiable.
- El equipo encriptador debe ser portátil y operable por una sola persona.
- El sistema debe ser simple, sin requerir memorización de muchas reglas ni generar esfuerzo mental.

Antes de continuar con la historia de la criptografía, veamos algunos elementos básicos de los métodos clásicos que servirán para entender la evolución de estos temas. Comencemos por los métodos simétricos.

Por ejemplo, supongamos que consideramos del alfabeto español solamente las letras minúsculas para no complicar el tema, o sea, que excluimos a todos los demás caracteres, y lo permutamos en forma arbitraria tal como aparece en la segunda línea del siguiente cuadro:

a	b	c	d	e	f	g	h	i	j	k	l	m	n	ñ	o	p	q	r	s	t	u	v	w	x	y	z
x	z	y	w	v	u	r	s	t	p	q	o	ñ	m	n	k	i	l	f	e	d	h	g	j	b	a	c

Entonces, si nuestro mensaje es "vamos al cine", lo transformamos según la tabla anterior. A la letra "v" le corresponde la letra "g" en la segunda fila, así que repitiendo esto para todo el mensaje llegamos a "gxñkexoytmv" (nótese que en nuestro alfabeto reducido no tenemos espacios, así que el texto encriptado tampoco los tiene). En este caso la clave es la permutación, o sea, la tabla anterior. Para desencriptarlo basta ver en la segunda fila de la tabla qué caracteres le corresponden a cada uno de los elementos del texto codificado. Por ejemplo, la letra "g" en la segunda fila proviene de la letra "v", la "x" de la "a", etc., y por lo tanto el texto decodificado comienza con "va". Pero la idea básica es numerar las letras del alfabeto, transformando así los mensajes en secuencias numéricas a las cuales se les aplican diversas operaciones. De este modo obtenemos:

a	b	c	d	e	f	g	h	i	j	k	l	m	n	ñ	o	p	q	r	s	t	u	v	w	x	y	z
0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26

Aunque la criptografía se remonta a miles de años atrás, uno de los primeros métodos documentados fue el del emperador Julio César y consistía en desplazar las letras de un texto tres lugares. En Halsall (1998) se lee lo siguiente:

> Si tenía que decir algo confidencial, lo escribía usando el cifrado, esto es, cambiando el orden de las letras del alfabeto, para que ni una palabra pudiera entenderse. Si alguien quiere decodificarlo, y entender su significado, debe sustituir la cuarta letra del alfabeto, es decir, la "d" por la "a", y así con las demás (Suetonio, *Vida de los Césares*).

Por ejemplo, la letra "a" tiene el valor "0", y si le sumamos 3 vemos que se transforma en la "d". Ahora, ¿qué sucede cuando queremos desplazar la letra "y", cuyo valor es "25"? Al sumarle 3 obtenemos 28 pero solamente tenemos hasta el número 26. Esto es similar a los relojes, pues cuando son las 22 hs y nos preguntamos qué hora será dentro de tres horas, tendríamos 22 + 3 = 25 pero el día solo tiene 24 horas. Entonces, decimos que es la 1 am, o sea que le restamos 24 al 25 que recién obtuvimos. Matemáticamente esto significa que usamos aritmética modular, o sea que todas las operaciones las hacemos en los relojes "módulo 24", consistente en dividir al resultado por 24 y tomar el resto de la división entera. Volviendo a Julio César, teníamos 28 como resultado de transformar la letra "y", y nuestro módulo es 27, igual a la cantidad de letras del alfabeto español (recuerden que empezamos a contar de 0). Entonces, al dividir 28 por 27, el resto entero resulta ser igual a 1, y esa es la letra "b".

Una imagen geométrica de lo anterior es pensar que escribimos las letras sobre un círculo y nos desplazamos

tres posiciones a lo largo de él. Cuando se acaba el círculo volvemos a recorrerlo desde el principio, o sea que "eliminamos" las vueltas completas, y eso es exactamente lo que hace la aritmética modular.

Naturalmente que el número 3 puede cambiarse por cualquier otro.

Los anteriores son ejemplos de métodos monoalfabéticos, pues modificamos los caracteres de a uno. El inconveniente obvio es que las letras se transforman siempre de la misma manera. Por ejemplo, la "a" de "vamos" se transformó en "x" en el método de permutaciones, lo mismo que la "a" de "al". En otras palabras, se conservan las frecuencias de aparición de las letras, y eso nos conduce al criptoanálisis, que es el arte de descifrar mensajes. Justamente una técnica clásica es hacerlo estudiando la frecuencia con la cual aparecen las letras, los digramas (combinaciones de dos letras) y los trigramas (combinaciones de tres letras). En el idioma español la letra "e" es la más frecuente, así que si en un mensaje encriptado vemos que la letra más frecuente es la "x", una hipótesis que se debe analizar es si esa "x" realmente puede ser una "e" en el mensaje decodificado. Esas hipótesis deben corroborarse mediante el estudio de los digramas y trigramas, que a su vez tienen frecuencias propias en los distintos idiomas, y sabiendo que muchas combinaciones no existen, aunque estas evolucionan con el tiempo. Por ejemplo, el trigrama "www" no existía en castellano antes de la aparición de Internet.

Entonces, queda claro que el inconveniente grave de estos métodos simples es que conservan las frecuencias, por lo que son fácilmente atacables.

Uno más sofisticado, que perduró hasta la Segunda Guerra Mundial, es el de Blaise de Vigenère, diplomático, criptógrafo

y químico del siglo XVI que publicó el libro *Traicté des chiffres*, en 1585, aunque hoy en día se le atribuye a Giovan Battista Bellaso, quien lo describió en un libro de 1553. La idea es usar una palabra como clave. Por ejemplo, tomemos "vaso", que numéricamente es "22 0 19 15", y nuestro mensaje "vamosalcine", que se representa como "22 0 12 15 19 0 11 2 8 13 4". Procedamos a sumar con módulo 27 los números del mensaje con los números de la clave, y cuando esta se "termina" la volvemos a reutilizar. En una tabla tendríamos:

v	a	m	o	s	a	l	c	i	n	e	Texto original
22	0	12	15	19	0	11	2	8	13	4	Escrito numéricamente
22	0	19	15	22	0	19	15	22	0	19	Clave repetida hasta el final del texto
44	0	31	30	41	0	30	17	30	13	23	Suma normal (22 + 22 = 44; 12 + 19 = 31)
17	0	4	3	14	0	3	17	3	13	23	Suma módulo 27 (44 – 27 = 17; 31 – 27 = 4)
q	a	e	d	ñ	o	d	q	d	n	w	Texto encriptado según los números de la fila anterior

En definitiva "vamosalcine" se transformó en "qaedñodqdnw".

Un método más seguro –al menos para esa época– es el llamado *autokey* (autoclave), inventado por Gerolamo Cardano (matemático, físico y astrólogo italiano del siglo XVI) y luego perfeccionado por Blaise de Vigenère. Los pasos son los siguientes:

- Elegir el texto que se va a encriptar y una palabra clave, como en el método de Vigenère.
- Escribir la palabra clave encima del texto que se va a encriptar, y a continuación comenzar a escribir el texto que pasará a ser la nueva clave a utilizar (de ahí el nombre "autoclave" pues el mismo texto sirve para

cifrar). Seguir así hasta tener tantas letras como tiene el texto original.

- Ahora se necesita una *tabula recta*, atribuida a Johannes Trithemius, un monje alemán del siglo V.[1]

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
Ñ	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ
P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O
Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P
R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q
S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R
T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S
U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T
V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U
W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V
X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W
Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X
Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	Ñ	O	P	Q	R	S	T	U	V	W	X	Y

- Para encriptar el texto se ubica la fila correspondiente a la primera letra y la columna con la primera letra

1. Véase "Cifrado de Vigenère", en <www.es.wikipedia.org>.

de la clave. En su intersección está la letra correspondiente al texto encriptado.

- Y así se sigue con todo el texto.

Supongamos, por ejemplo, que el texto que se va a encriptar es nuevamente "vamos al cine" y la clave es "vaso".

V	A	S	O	V	A	M	O	S	A	L
V	A	M	O	S	A	L	C	I	N	E
Q	A	F	D	Ñ	A	W	Q	A	N	O

Entonces, según el método original de Vigenère, "vamos al cine" usando la clave "vaso" se transformaba en "qaedñodqdnw", y según el *autokey* el cifrado es "qafdñawqano".[2]

El método de Vigenère fue quebrado mediante una técnica desarrollada originalmente por Charles Babbage (matemático y científico del siglo XIX considerado el padre de la computación por sus inventos), pero dado que los ingleses lo conservaron como un secreto militar, fue luego atribuido al oficial alemán Friedrich Kasiski en 1863. Su idea fue buscar palabras repetidas en el texto encriptado que ocurren cuando hay palabras repetidas en el texto original y, además, la clave coincidió en su posición en ambos casos. Dado que la distancia entre palabras repetidas es un múltiplo de la longitud de la clave, su excelente propuesta fue buscar distintas repeticiones y calcular el máximo común divisor de estas para estimar un múltiplo cercano a la longitud de

2. Hay diversos programas interactivos en la Web que implementan estos métodos; por ejemplo, en el sitio Crypto Corner.

la verdadera clave. La longitud correcta de la clave será este número o alguno de sus factores primos. Hecho esto, hay software para calcular todas las combinaciones de claves con esa longitud hasta lograr el descifrado correcto. También se puede dividir el texto original en bloques de longitud igual a la de la clave y aplicar análisis de frecuencias, como cuando se ataca el método de Julio César.

A finales del siglo XIX la criptografía estaba en un serio problema. Los trabajos de Babbage y Kasiski habían servido para quebrar el cifrado de Vigenère, y no se tenía ninguna alternativa viable, sobre todo por el advenimiento de la radio, inventada por el físico italiano Guillermo Marconi. Inicialmente se creyó que el alcance de las transmisiones radiales era muy limitado pues se suponía que el horizonte era un límite infranqueable. Pero el 12 de diciembre de 1901 Marconi logró transmitir un mensaje a 3500 kilómetros de distancia, algo que en ese momento resultaba increíble. La explicación se consiguió recién en 1924, cuando se descubrió la existencia de la ionósfera, la cual actúa como un espejo para las ondas radiales y las dirige hacia la Tierra, donde vuelven a rebotar, etc. Ese invento revolucionario atrajo inmediatamente la atención de los militares, pues se abrían posibilidades inimaginables, como conectar dos unidades en forma remota sin utilizar cables. Pero obviamente las transmisiones también serían capturadas por el enemigo y, entonces, la necesidad de encriptar los mensajes se transformó en un tema acuciante.

Durante la Primera Guerra Mundial no hubo ningún avance significativo pues simplemente se probaron variaciones de los métodos conocidos, ninguna de las cuales resistió el ataque de los criptoanalistas.

En 1914 el Almirantazgo británico creó la llamada *Room 40* (Habitación 40), que era la sección dedicada al criptoanálisis.

Uno de los desastres militares más impactantes tuvo lugar en 1918, o sea, cuando terminó la Primera Guerra Mundial. Los alemanes utilizaron ese año por primera vez un método de encriptación denominado ADFGVX. En junio de 1918 la artillería alemana estaba muy próxima a París y los Aliados necesitaban saber en forma perentoria los detalles del ataque que esperaban. Un criptoanalista francés, el teniente Georges Jean Painvin (1886-1980), formado en la École Polytechnique y miembro del Bureau de Chiffre, logró quebrar los mensajes encriptados de los alemanes. Uno en particular mostraba la ubicación de la artillería y pedía municiones con urgencia. Esos datos fueron entonces confirmados mediante reconocimiento aéreo y una semana después las tropas alemanas fueron masacradas.

Sin duda, en la época de la Primera Guerra Mundial Francia era el país más avanzado en términos de criptoanálisis. Auguste Kerckhoffs, aunque era holandés, había desarrollado sus actividades en Francia, donde publicó su tratado *Cryptographie militaire*, que tuvo gran influencia sobre la formación de los criptoanalistas. Por su parte, Alemania había entrado en la guerra sin nada comparable con el Bureau de Chiffre de Francia, y recién dos años después creó una organización para interceptar los mensajes de los Aliados. Una razón fue que los franceses, al ser invadidos por los alemanes, destruyeron en su retirada las líneas de comunicaciones terrestres. Esto obligó a Alemania a tener que usar comunicaciones radiales para todos sus movimientos militares, por lo que tenía mucho sentido para Francia interceptarlas.

Un detalle interesante es que los franceses perfeccionaron el llamado "análisis de tráfico", o "inteligencia radial". Este no involucraba técnicas de descifrado, sino que apuntaba a ubicar la fuente de una transmisión mediante la triangulación de las señales de radio y, por otra parte, la identificación de los operadores hechas en código Morse a través del estudio de sus cadencias de escritura. Por ejemplo, tanto las pausas como la velocidad son características de las personas, y es curioso que tuviéramos que esperar hasta la actualidad para utilizar el mismo tipo de ideas como parte de la autenticación remota. Veremos luego que el mismo enfoque resultó muy útil para quebrar mensajes en la Segunda Guerra Mundial.

Un hecho importantísimo fue el *affaire* del telegrama Zimmermann. Los británicos querían a toda costa que los Estados Unidos entrasen en la guerra, algo que era resistido en ese país por múltiples razones políticas. De hecho, pensaban que podían permanecer neutrales y mediar en el conflicto. Pero el año anterior, el 7 de mayo de 1915, hubo un hecho que conmovió al mundo: el hundimiento del transatlántico británico *RMS Lusitania* por parte del submarino alemán *U-20*, el cual lanzó sus torpedos mientras estaba sumergido y provocó la muerte de 1198 personas. Salir o no a la superficie antes de atacar era un tema crucial, y el recién nombrado ministro de Relaciones Exteriores de Alemania, Arthur Zimmermann, le aseguró a los Estados Unidos que se cumpliría estrictamente con la prohibición de repetir un episodio semejante para evitar hundir barcos mercantes. Sin embargo, a principios de 1917, los militares querían persuadir al Kaiser de que había que desatar una guerra sin limitaciones pues estaban convencidos de que sus submarinos

eran indetectables si permanecían sumergidos. Esto, obviamente, implicaba incumplir el compromiso adoptado por Zimmermann. La idea era que los doscientos submarinos alemanes podrían afectar severamente las líneas de suministros de los ingleses y forzar la rendición. Pero comenzar con esa política podría conducir a que los Estados Unidos –que habían pasado por alto el ataque criminal contra el *Lusitania*– revieran su decisión de no entrar en la guerra. Por ello, Zimmermann diseñó una estrategia consistente en lograr que México atacase a los Estados Unidos para recuperar Arizona, Nuevo México y Texas con la ayuda financiera y militar de Alemania, y convencer a Japón para que lo hiciera desde el Pacífico. Naturalmente, la idea era crearles a los Estados Unidos suficientes conflictos en su propio territorio como para que se olvidasen de intervenir en el escenario europeo.

Zimmermann redactó su maquiavélico plan y necesitaba mandarlo encriptado al embajador alemán en México para que se lo transmitiese al presidente de ese país. Alemania utilizaba cables transatlánticos submarinos como su instrumento más seguro de comunicaciones, y por ese motivo los británicos los sabotearon el día que comenzó la guerra. En consecuencia, dada su importancia, el mensaje debió ser enviado por dos medios distintos: por telegrama vía Suecia y a través de un cable propiedad de los Estados Unidos. Eso condujo a que el mensaje encriptado fuese interceptado por los británicos, quienes lograron desencriptarlo, primero de forma parcial pero obteniendo suficiente información como para darse cuenta de su importancia, y de que ineludiblemente llevaría a que los Estados Unidos entraran en la guerra. Sin embargo, el director de la Inteligencia Naval inglés pensó, como años

más tarde lo haría Churchill en la Segunda Guerra Mundial, que si hacía llegar la información al presidente Wilson, y los Estados Unidos reaccionaban en forma visible, los alemanes se darían cuenta de que los ingleses habían logrado descubrir su método de cifrado. Además, dado que los alemanes estaban usando líneas de comunicación estadounidenses que pasaban por Inglaterra, si los ingleses ponían en evidencia que habían interceptado el mensaje de Zimmermann también mostrarían que estaban espiando las comunicaciones de sus aliados. Eso a su vez llevaría a que Alemania se viera forzada a desarrollar métodos diferentes, y de ese modo Inglaterra perdería la capacidad vital de quebrar sus comunicaciones. De todas maneras, faltaban solo un par de semanas para que los alemanes lanzaran los ataques de sus submarinos sumergidos, lo cual sería un motivo más que suficiente para que los Estados Unidos tomaran la decisión de entrar en la contienda. Pero nada de eso sucedió, pues el presidente Wilson le anunció al Congreso que mantendría la política de neutralidad.

El telegrama encriptado de Zimmermann fue enviado al embajador alemán en Washington para que lo retransmitiese al de México, luego de hacerle unos cambios obvios, como remover las instrucciones recibidas. La Inteligencia británica asumió que el envío a México se haría mediante los canales telegráficos comerciales comunes para evitar sospechas dado que era un mensaje entre dos países neutrales, y por lo tanto debería haber existido una copia en la oficina de telégrafos de la Ciudad de México. Esa copia fue obtenida por un agente llamado Mr. H. y, casi por milagro, lo habían encriptado con un método antiguo llamado "cifrado 13040", o sea que descifrarlo no suponía exponer lo que realmente habían logrado: quebrar el cifrado 0075. Este telegrama ya descifrado, o sea,

la versión "mexicana" del texto de Zimmermann, fue remitido por la Inteligencia británica a Arthur Balfour, secretario de Estado de Asuntos Extranjeros inglés, quien a su vez lo entregó al embajador estadounidense. El presidente Wilson lo recibió cuatro días después, y fue entregado a los medios de comunicación el 1 de marzo de 1917. De todos modos, existía la sospecha de que todo fuera una maquinación de los ingleses para que los Estados Unidos declarasen la guerra a Alemania. Sorprendentemente, Zimmermann admitió su autenticidad en una conferencia de prensa en Berlín el 3 de marzo de 1917. Esto condujo a un cambio en la opinión pública estadounidense respecto de la neutralidad.

Naturalmente, el gobierno alemán inició una investigación para dilucidar cómo un secreto de tal importancia se había filtrado, pero cayeron en la trampa de la Inteligencia británica al concluir que la traición había ocurrido en México. El resultado final de esta increíble historia fue que los Estados Unidos le declararon la guerra a Alemania, algo que tres años de diplomacia no habían conseguido y que finalmente sucedió gracias al esfuerzo de dos criptoanalistas.

Una pregunta obvia es si existe algún método inquebrable. La respuesta es afirmativa. Se trata del "anotador de única vez" [*one time pad*] que describiremos a continuación. Consideremos nuevamente la permutación que utilizamos antes:

a	b	c	d	e	f	g	h	i	j	k	l	m	n	ñ	o	p	q	r	s	t	u	v	w	x	y	z
x	z	y	w	v	u	r	s	t	p	q	o	ñ	m	n	k	i	l	f	e	d	h	g	j	b	a	c

Para encriptar el mensaje "vamos al cine", que dio como resultado "gxñkexoytmv", podemos usar obviamente otras permutaciones, como:

a	b	c	d	e	f	g	h	i	j	k	l	m	n	ñ	o	p	q	r	s	t	u	v	w	x	y	z
x	z	y	w	v	u	r	s	t	p	q	o	ñ	m	n	k	i	l	f	e	d	h	g	j	b	a	c

Ahora supongamos que elegimos tantas permutaciones como letras tenga el mensaje y que transformamos la primera letra según la primera permutación, la segunda según la segunda, etc. En nuestro caso la primera letra ("v") se transforma en "g" como antes, pero la segunda, "a", se transforma en "h", y así siguiendo. Claude Shannon demostró en 1949 que si las claves del anotador de única vez no se vuelven a utilizar, el método posee la propiedad del secreto perfecto, lo que significa que no se puede quebrar. El problema esencial es que necesitamos disponer de claves con longitudes iguales a la de los mensajes que se van a transmitir, y de ahí surge una cuestión obvia: ¿cómo distribuimos las claves en forma segura? Trataremos este tema cuando lleguemos a la criptografía de clave pública.

En definitiva, el método del anotador de única vez se puede considerar como un cifrado de Vigenère con una clave de longitud igual al texto que se va a codificar.

Como es sabido, todos los caracteres (letras, números, caracteres especiales, etc.) se representan internamente en una computadora mediante conjuntos de ocho bits (números binarios, o sea, 0, 1), que se denominan *bytes*, utilizando el código llamado ASCII (sigla de *American Standard Code for Information Interchange*). Por ejemplo, la letra "d" se representa por 00001101.

¿Cuál es la relación de la representación de los caracteres mediante dígitos binarios con el anotador de única vez y la teoría del secreto perfecto de Shannon? Para ello repasemos algo de lógica, en particular, la operación de la disyunción

denominada "xor". Si "0" representa falso y "1" verdadero, la tabla de verdad es:

x	y	xy
0	0	0
0	1	1
1	0	1
1	1	0

Si x = "esta noche iremos al cine" e y = "esta noche iremos al teatro", la primera fila indica que si ambas proposiciones son falsas el resultado es falso, la segunda y tercera muestran que si una de las proposiciones es verdadera el resultado es verdadero, y la última da falso pues no podemos ir al cine y al teatro simultáneamente. Supongamos que queremos encriptar la letra D = 01000100 mediante la clave K = 10011101, de igual longitud que nuestro "mensaje", utilizando la tabla anterior. Entonces,

0	1	0	0	0	1	0	0	D = mensaje original
1	0	0	1	1	1	0	1	K = clave
1	1	0	1	1	0	0	1	Encriptado = DK

¿Qué sucedería si un espía accede al mensaje original, sea antes de encriptarlo o luego de desencriptarlo, y también al texto cifrado? Entonces, puede realizar la siguiente operación:

0	1	0	0	0	1	0	0	D = mensaje original
1	1	0	1	1	0	0	1	Encriptado = DK
1	0	0	1	1	1	0	1	K = clave

O sea que logra recuperar la clave que se utilizó, y por ello esta no puede ser empleada nuevamente.

Aquí es necesario discutir con más profundidad el tema de la generación de números aleatorios pues es una clave del espionaje que se lleva a cabo sobre los sistemas informáticos. De hecho, el ejemplo anterior nos muestra que lo que necesitamos es generar sucesiones aleatorias de dígitos binarios para utilizar el anotador de única vez, o algo que se le parezca lo más posible.

Un caso conocido del uso del anotador de única vez es el teléfono rojo Moscú-Washington ante el peligro de guerra nuclear, por lo que ambos gobiernos acordaron una clave secreta de gran longitud que se usaría solo en caso de peligro inmediato de conflicto. Otro ejemplo es justamente Rusia, que comunica a sus embajadas con Moscú mediante claves del anotador de única vez que se distribuyen por correo diplomático.

Ahora bien, en la vida real no disponemos de correo diplomático o métodos similares para distribuir las claves, y por eso utilizamos generadores de números aleatorios. Pero lo ideal sería que las claves sean perfectamente aleatorias, algo que se puede lograr solo midiendo magnitudes físicas como la temperatura, las variaciones de voltaje, la precipitación de los rayos cósmicos, el decaimiento radioactivo, los movimientos de un ratón, los sonidos de un aparato, o mediante la criptografía cuántica.

De hecho en una computadora de 32 bits (o sea, las que utilizan cuatro *bytes* para representar los números reales, instrucciones, etc.) cualquier mecanismo de generación de números aleatorios tiene a lo sumo un ciclo de elementos distintos, y por eso se denominan "números seudoaleatorios". Por lo tanto, las secuencias de números generados es hasta un punto "predecible", y de ahí la conveniencia de buscar métodos alternativos. En el caso de computadoras de 64 bits el ciclo es naturalmente de 2^{64}.

Las computadoras de 32 bits manejan enteros en el rango de 0 a 4.294.967.295, y en las de 64 bits esto cambia al rango entre 0 y 18.446.744.073.709.551.615. Estos números pueden parecer gigantescos, aunque no lo son desde el punto de vista de la criptografía.

Pero sigamos con la historia. Habíamos mencionado que Leon Alberti creó el primer dispositivo de encriptación por hardware mediante dos discos concéntricos que implementaba el método de Vigenère, quebrado como vimos por Babbage y Kasiski. Sin embargo, la idea prosperó y fue el origen de los desarrollos que condujeron a la máquina Enigma –realmente hubo muchos modelos de este dispositivo de cifrado, desde la Enigma A de 1923 hasta el modelo KD de 1944, el D de 1927 de la *Reichswehr* (fuerzas armadas de Alemania de 1919 a 1935), el modelo 1 de la *Wehrmacht* (fuerzas armadas de Alemania de 1935 a 1946) y *Luftwaffe* (fuerza aérea en la época nazi) de 1932, la serie M1, M2, M3 y M4 de la *Kriegsmarine* (Armada del III Reich entre 1935 y 1945), y variaciones hechas en Suecia y en Suiza (Reuvers y Weierud, 2009)–. Esta comenzó con el desarrollo hecho por un inventor alemán, Arthur Scherbius, quien quiso reemplazar los métodos manuales y deficientes usados en la Primera Guerra Mundial por tecnología acorde con el siglo XX. Debido a su formación como ingeniero eléctrico desarrolló la "versión eléctrica" de los discos de Alberti y la bautizó Enigma. Sus partes eran:

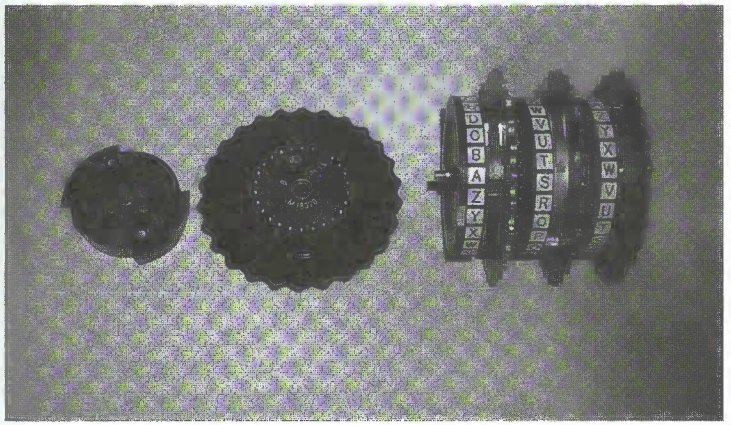
- Tres discos llamados "rotores" [*Walzen*] conectados por cables.
- Un teclado para escribir los textos que se iban a encriptar.
- Una unidad "mezcladora" que transformaba cada letra del texto original en uno encriptado.
- Un tablero con lámparas que al encenderse indicaban cuál era la letra encriptada.
- Un anillo de goma resistente con 26 números o letras y una muesca, cuya función era la de cambiar la conexión del cableado con respecto a los rotores.
- Un reflector [*Umkehrwalze* o *UKW*] que efectuaba las transformaciones de las letras de a pares. Por ejemplo, si la letra H se transformaba en X con una cierta clave, el reflector transformaba la X en H. Esto permitía encriptar y desencriptar fácilmente. Pero nunca la encriptación de una letra podía derivar en esa misma letra, y esa era una debilidad que se usó para criptoanalizar a Enigma (al respecto, cabe mencionar que al finalizar la Segunda Guerra Mundial los rusos desarrollaron una máquina cifradora llamada Fialka, basada en las ideas de la Enigma, pero mucho más avanzada. En particular tenía diez rotores y podía resultar que la encriptación de una letra fuera ella misma).

Los rotores funcionaban como el odómetro de un automóvil, o sea que cada vez que se introducía una letra el rotor giraba y, al completarse una vuelta, el siguiente rotor giraba una posición, etc. Esto se lograba mediante una muesca, pero también inventaron alternativas con un número distinto de muescas. Por ejemplo, la Enigma-G tenía 11, 15

y 17 muescas en cada uno de sus rotores, lo que hacía que el rotor vecino avanzara antes de que el anterior diese una vuelta completa.

Parte de una clave para una sesión era por supuesto la posición inicial de los rotores. Dado que usaban un alfabeto de 26 letras –o sea sin "ñ"–, la cantidad de claves iniciales posibles era 26 x 26 x 26 = 17.576 posibilidades, lo cual no era suficientemente complejo. Una alternativa para aumentar la seguridad era agregar más rotores –algo que hizo la Marina alemana como veremos más adelante– pero en ese momento se consideró que hacerlo incrementaría el peso de la máquina, que ya era de 12,5 kilos.

Figura 2.1
Rotores de la máquina Enigma



Fotografía: cortesía del Enigma Museum (enigmamuseum.com).

Scherbius tuvo dos ideas: la primera fue disponer de distintos rotores y hacerlos intercambiables, y la segunda consistió en añadir un tablero de conexiones [*Steckerbrett*] (usado solo por el Ejército), que permitía cambiar las letras a través de cables. Por ejemplo, si el operador escribía una "a" un cableado podía transformarla en "b" o en "j", etc., según cómo se conectaran.

Figura 2.2
Tablero de conexiones

Fotografía: cortesía del Enigma Museum (enigmamuseum.com).

Figura 2.3
Máquina Enigma de cuatro rotores completa

Fotografía: cortesía del Enigma Museum (enigmamuseum.com).

Consideremos ahora la complejidad para quebrarla:

- Teníamos 17.576 posibilidades por los tres rotores.
- Los rotores podían intercambiarse, así que, si los numeramos, las posibilidades son 123, 321, 312, 132, 213, 231, o sea, seis en total.
- El cableado permitía conectar seis pares de letras de las 26, y eso da 100.391.791.500 alternativas, o diez pares

de letras usando diez cables (esa era la cantidad que se solía utilizar), en cuyo caso resulta 150.738.274.900.000. Aunque se podían usar hasta trece cables, el máximo de combinaciones se obtiene para once, en cuyo caso resultan ser 205.552.193.100.000. No se sabe por qué los alemanes no eligieron esa opción.

- El número de alternativas es igual a la multiplicación de esos números, o sea, 10.586.916.764.424.000 con seis cables, y con diez cables resultan 2.649.375.919.642.400.000 posibilidades.

Obviamente es imposible considerar el quiebre de los códigos probando todas las combinaciones posibles.

A esta estimación de la complejidad hay que agregarle las 666 (26 x 26) variaciones posibles de los anillos. En términos más modernos esto equivale a un cifrado de 84 bits.

Scherbius y su socio, Richard Ritter, fundaron una pequeña empresa llamada Chiffrienmaschinen Aktien Gesellschaft pero no fueron exitosos desde el punto de vista comercial debido, por un lado, al alto costo de la máquina, y, por otro, a la falta de conciencia acerca de la seguridad, un tema que todavía persiste en muchas organizaciones. Todo ello a pesar de que presentaron su invento en el congreso de la Unión Postal Internacional y de que el director del Instituto de Criminología de Viena, Siegfried Türkel escribió en forma muy extensa y elogiosa acerca de la nueva máquina. Lo mismo aconteció con otros equipos de cifrado, como el de Hugo Alexander Koch en Holanda (1919), o el de Arvid Damm en Suecia. Pero fue el estadounidense Edward Hebern el primero en usar la idea de los rotores en máquinas de cifrado. Su paso inicial fue conectar en 1915 dos

máquinas de escribir eléctricas mediante veintiséis cables dispuestos al azar de modo tal que al oprimir una letra en la primera máquina otra aparecía impresa en la segunda. Dado que los cables permanecían fijos durante el proceso este método era una simple transformación monoalfabética, y por lo tanto no resistía el más mínimo análisis. En 1917 Hebern hizo los planos de un sistema de rotores y fabricó el primero al año siguiente. Luego, en 1921, publicitó un mecanismo "inquebrable" mediante un mensaje cifrado que fue quebrado inmediatamente por una criptoanalista de la Marina. A pesar de esto, la Marina, que estaba procurando un sistema para las comunicaciones secretas, se interesó por el aparato pero poniendo como condición que lo perfeccionara. Sin embargo, todo terminó en un gran fracaso comercial y solamente unas pocas máquinas permanecieron en servicio hasta 1942.

Por su parte, los militares alemanes no se mostraban interesados, pues todavía en esa época no se habían dado cuenta de que el telegrama de Zimmermann había sido descifrado por los británicos. El panorama cambió gracias a dos documentos ingleses. El primero fue el libro de Winston Churchill, *La crisis mundial 1911-1918*, publicado en 1923, donde se relataba que, luego del hundimiento del crucero alemán *Magdeburg* en el mar Báltico, los soviéticos habían encontrado el cuerpo de un oficial alemán que tenía los códigos de cifrado y de señales de la Armada. Con esos datos el Almirantazgo ruso pudo descifrar buena parte de los mensajes navales alemanes y, curiosamente, los rusos consideraron que debían compartir esa información con los ingleses. A partir de la entrega de ese material, los ingleses comenzaron a descifrar los mensajes alemanes. El libro de Churchill

hizo tomar conciencia a los militares alemanes acerca de la vulnerabilidad de sus comunicaciones.

Ese mismo año, la Armada Real publicó la historia oficial de lo que había sucedido en la Primera Guerra Mundial, explicando cómo la intercepción y el criptoanálisis de las comunicaciones alemanas habían sido determinante para que las fuerzas aliadas tuvieran una clara ventaja sobre el enemigo.

Como conclusión, en 1926 Alemania decidió apoyar fuertemente el uso de la máquina Enigma, sin duda el método más avanzado de criptografía de la época. Para ello el Almirantazgo contactó secretamente a la empresa y le encargó a un oficial de extrema confianza, Johannes Möller, que transportara una Enigma desde Berlín a Kiel. Así, la empresa fundada por Scherbius, quien ya había fallecido, le pudo vender finalmente unas 30.000 máquinas al gobierno alemán, luego de hacerle algunas modificaciones solicitadas por la Marina y aceptadas por el Ejército.

Se fabricaron distintos modelos de Enigma, tanto para aplicaciones comerciales como para los ferrocarriles, la *Luftwaffe*, etc.

Se puede visitar <www.cryptomuseum.com>, así como numerosas referencias en la Web, para ampliar esta información.[3] En la próxima sección veremos que la *Kriegsmarine* desarrolló la Enigma M4, con cuatro rotores, como respuesta

3. En Internet se encuentran varios programas que simulan perfectamente el funcionamiento de los distintos modelos de Enigma. Por ejemplo, Enigma Simulator v7.0 y Enigma World Code Group.
En el sitio <www.cryptomuseum.com> hay muchas referencias, imágenes y links a otros simuladores, museos, etc. En YouTube, bajo el título "Enigma Machine Simulator-Nazi Code Machine", puede verse un video que explica el funcionamiento de un simulador.
Un film muy raro que muestra la operación de una máquina Enigma puede verse en <enigmamuseum.com/enigop2.wmv>.

a las suposiciones acerca del posible quiebre de las encriptaciones realizadas con las máquinas de tres rotores.

Antes de analizar el quiebre de la Enigma de tres rotores vale la pena mencionar que los Estados Unidos tuvieron también máquinas de encriptación, como la M-209 diseñada por Boris Hagelin, nacido en Azerbaiyán de padres suecos, quien estudió ingeniería mecánica en Estocolmo. El padre de Hagelin invirtió en la compañía de Damm y utilizó sus patentes. La M-209 interesó a los militares estadounidenses, pero fue quebrada por los alemanes pues era mucho menos segura que la Enigma. Esa máquina fue usada por los italianos (C-38m era su identificación), y por ello los ingleses sabían todo acerca de la situación de los suministros para el mariscal Rommel en la campaña del norte de África, y a su vez los alemanes conocían los detalles de los ataques estadounidenses en términos de objetivos y fechas, durante 1942 y 1943.

Durante la guerra Hagelin emigró a Suiza y fundó una empresa que todavía existe llamada Crypto AG, de larga y complicada historia.

CÓMO SE QUEBRÓ LA MÁQUINA ENIGMA

Francia fue el país más afectado por la Primera Guerra Mundial pues parte de su territorio había sido ocupado por los alemanes, varios millones de hombres habían muerto en la contienda, la supuestamente inexpugnable Línea Maginot había colapsado y la economía estaba exhausta luego del esfuerzo bélico. Por esos motivos su Estado Mayor era particularmente sensible a los informes de inteligencia que

mostraban que Alemania estaba violando los términos del Tratado de Versalles, es decir, reconstruyendo su industria, rearmando sus fuerzas y entrenándolas en la Rusia soviética. Así es como enviaron a Berlín al capitán Gustave Bertrand, del Segundo Bureau del Estado Mayor General, dado que tenía mucha experiencia en inteligencia de radio, hablaba correctamente el alemán y desde 1926 estaba siguiendo de cerca los informes secretos que mostraban que habían comenzado a utilizar una nueva máquina de cifrado que producía textos encriptados supuestamente no quebrable. Los franceses comenzaron a trabajar sobre el tema, pero recién redoblaron sus esfuerzos en 1929, cuando se confirmó que tanto el Ejército como la Armada de Alemania disponían de la Enigma. Sin embargo, en principio era difícil utilizar esos informes porque las máquinas Enigma se vendían libremente, así que era lógico suponer que la versión militar debía incluir otros elementos que le confirieran mayor seguridad.

Bertrand fue nombrado director de la Sección D de la Inteligencia francesa en 1930, y en el otoño de 1931 el capitán Henri Navarre le informó que había sido contactado por un joven alemán que trabajaba en la Oficina de Cifrado de la *Reichswehr* que contaba con información muy importante para los franceses. Su nombre era Hans-Thilo Schmidt, había nacido en Berlín en 1888 y era el hermano menor del general Rudolf Schmidt. Tanto su identidad como su posición fueron investigadas a fondo para garantizar la confiabilidad de la información que podía suministrar, aunque por un tiempo persistió la duda de que respondiera a medidas de contrainteligencia.

Bertrand decidió organizar reuniones con Schmidt, quien quería tomar el máximo de precauciones y llevarlas a cabo en terreno neutral y no lejano pues solo disponía de los

fines de semana para hacerlas, dado que debía presentarse los lunes a trabajar. La primera tuvo lugar en Verviers, un pueblo belga, y allí Schmidt llevó una cantidad importante de documentos que debían fotografiarse pues tenía que devolverlos de inmediato. Como consecuencia del éxito de esa primera reunión se organizaron otras en Berna, Basilea, Copenhague, Zúrich, etc. Debido a que el material que entregaba era sumamente relevante le hicieron un pago significativo, lo enrolaron como informante y le dieron un nombre en clave: Asché. Sin embargo, persistían las dudas acerca de sus verdaderas motivaciones.

Lo que se determinó fue que había peleado en la Primera Guerra Mundial, pero luego había sido dado de baja por las restricciones impuestas por el Tratado de Versalles. Luego se dedicó a los negocios: fundó una fábrica de jabones que tuvo que cerrar por la depresión económica y la altísima inflación, por lo que él y su familia quedaron en la ruina, y le tuvo que pedir ayuda a su hermano, quien lo ubicó en la *Chiffrierstelle*, la oficina responsable de las comunicaciones secretas. Su hermano Rudolf había escalado posiciones en el Ejército: fue general y jefe del Cuerpo de Señales. Más aún, fue la persona que aprobó el uso de la máquina Enigma en el Ejército. En otras palabras, sus motivaciones eran primordialmente de índole psicológica: le tenía envidia a su hermano, estaba resentido con su país, que lo había dejado de lado y sumido en problemas económicos. Suministrar información secreta a Francia sobre Enigma era una venganza perfecta.

La información que les proporcionaba era muy completa y a partir de ella se pudo deducir lo que faltaba, lo que permitió que los Aliados quedaran en condiciones de cons-

truir una réplica de la Enigma. Pero también estaba claro que tener una Enigma no era suficiente pues la gran complejidad de los códigos producidos con una clave dada hacía imposible quebrar los mensajes. De hecho, un memorándum alemán de la época afirmaba que había que asumir que el enemigo tenía a su disposición una Enigma. Y como los criptoanalistas franceses eran conscientes de este hecho, ni siquiera se preocuparon por construir una réplica.

La documentación provista por Schmidt también explicaba la estructura de los libros de claves para la Enigma. Una vez por mes los operadores recibían un libro de códigos donde se especificaba para cada día lo siguiente:

- El cableado, por ejemplo A-W B-Y P-D, etc. Esto implicaba transformar la "a" en "w" mediante un cable de conexión, la "b" en "y", etc.
- La combinación de los rotores, por ejemplo 1-3-2, que significaba poner el rotor 1 en la posición 1, el rotor 3 en la posición 2 y el rotor 2 en la posición 3.
- La posición inicial de los rotores, por ejemplo C-J-W. El operador debía mover el rotor 1 hasta que apareciese la letra "c", el rotor 2 hasta llegar a la "j" y el rotor 3 hasta que mostrase la "w".

Con esto se comenzaba a operar. Una posibilidad era usar esta clave durante todo un día, pero lo que más necesitaban los criptoanalistas era tener una gran cantidad de texto encriptado con la misma clave. Por ello, inventaron algo ingenioso aunque creaba una vulnerabilidad: se generaba una nueva combinación para la posición inicial de los rotores, por ejemplo H-X-A, y se la encriptaba con la clave original C-J-W por duplicado por si había errores de transmisión. Por ejemplo, si H-X-A se transformaba mediante la clave C-J-W en Z-P-Q, lo que se transmitía en el siguiente mensaje era como seis primeras letras ZPQZPQ,; el operador la descifraba obteniendo H-X-A y, sin cambiar ni el cableado ni la combinación de los rotores, procedía a descifrar el cuerpo del mensaje. Pero la segunda vez que se encriptaba H-X-A lo que se lograba era consecuencia de que los rotores habían avanzado, o sea que si H-X-A se transformaba en Z-P-Q con la clave C-J-W, la segunda terna H-X-A podía derivar en R-A-B. En definitiva, lo que se transmitía era ZPQRAB: la "z" era la encriptación de la primera letra de la clave con la posición inicial de los rotores y la "r" era la de esa primera letra pero luego de que los rotores hubiesen avanzado una posición.

Sin embargo, y siempre pensando que los servicios secretos de los Aliados intentaban quebrar sus mensajes, se introducían diversos cambios en forma sistemática. Por ejemplo, desde octubre de 1936 el número de cables variaba entre cinco y ocho, pero desde comienzos de 1939 pasaron a usar entre siete y diez, y la cantidad nunca era constante.

Lo que permitió descifrar la Enigma fue el trabajo de los matemáticos polacos. De hecho, Francia y Polonia habían firmado un acuerdo de cooperación militar diez años atrás, así que cuando los franceses obtuvieron la documentación de Schmidt mandaron copias fotográficas a la organización polaca llamada Biuro Szyfrów, encargada del tema. Los polacos venían observando los avances germánicos en estos temas, pensaban igual que sus colegas franceses acerca de que el tema crucial residía en la complejidad del cifrado de la Enigma, pero tenían una presión mucho mayor para actuar debido al temor de una invasión.

Siempre se había pensado que los criptoanalistas debían ser lingüistas expertos en las estructuras de los lenguajes, pero con el advenimiento de las máquinas cifradoras los polacos consideraron que había que formar una nueva clase de expertos, lo cual era natural en un país con una larga tradición en lógica matemática. Por ese motivo organizaron un curso de criptografía para veinte matemáticos de la universidad de Poznań debido a que esa ciudad había sido parte de Alemania hasta fines de la Primera Guerra y, por ello, todos dominaban el idioma alemán. Del total solo tres que mostraron habilidades para quebrar textos cifrados fueron incorporados a la organización: Marian Rejewski, Jerzy Różycki y Henryk Zygalski. El primero era el más destacado; había estudiado Estadística, pero mostró mucho talento para descifrar mensajes tradicionales, hasta que su entrenamiento le permitió comenzar a dedicarse a la Enigma.

Mientras tanto Polonia había establecido puestos de escucha en Poznań, Krzesławice y Starogard Gdański para interceptar las transmisiones alemanas, y habían tenido éxito en descifrarlas hasta que de repente, en 1926, se encontraron con una categoría totalmente nueva de encriptación. Era la Enigma, y ello requería un esfuerzo urgente de la Inteligencia polaca. El primer paso fue comprar una a través de conexiones de inteligencia con una empresa de ingeniería eléctrica llamada AVA y con un industrial alemán, aunque a sabiendas de que la comercial debía diferir de la militar. El capitán Bertrand viajó a Varsovia con unos documentos muy importantes: el primero era el manual de instrucciones de la Enigma con cuatro dibujos de la máquina que mostraban su similitud con una que se había analizado en 1929, pero con la diferencia de que la nueva tenía el tablero de conexiones y también

un cableado distinto de los rotores. El segundo documento especificaba con qué frecuencia debían cambiarse los diversos componentes de la clave. Los rotores se intercambiaban cada tres meses, pero su posición y el cableado se modificaban diariamente (esto fue cambiando hasta que en 1942 se modificaban cada ocho horas). Más aún, Bertrand fue a Varsovia por segunda vez en 1932 llevando las claves para los meses de septiembre y octubre. Con toda esta información los polacos pudieron reconstruir la máquina y analizar cómo encontrar las claves.

Un primer paso fue el desarrollo del método de la grilla basado en que los cables cambiaban seis pares de letras pero dejaban catorce letras sin cambios; sin embargo, cuando los alemanes aumentaron el número de conexiones, este método perdió buena parte de su eficacia. Afortunadamente los criptoanalistas polacos descubrieron que la encriptación con las Enigma generaba ciclos. Entonces, construyeron un "ciclómetro" mediante la interconexión de dos máquinas. Por ejemplo, si una cierta clave producía J-W-H-J, esto era un ciclo de longitud tres. Lo que hicieron entonces fue construir una "base de datos" de gran tamaño, de modo tal que cuando aparecía un ciclo de un mensaje cifrado se podía ver cuáles eran las claves posibles que lo habían generado. En este punto Rejewski hizo un descubrimiento genial: ¡las longitudes de los ciclos dependían únicamente de la posición inicial de los rotores y no del cableado! En consecuencia, no era necesario explorar la gigantesca cantidad de combinaciones sino solo 105.456 posibilidades (seis combinaciones de rotores multiplicados por las 17.576 posiciones iniciales de las letras).

El ataque comenzaba entonces mediante la remoción del cableado y estudiando las primeras seis letras de los men-

sajes para determinar ciclos hasta definir la posición inicial de los rotores. Luego introducían textos cifrados que habían sido interceptados y probaban distintas combinaciones de claves hasta que aparecían jirones de palabras reconocibles. De este modo nada trivial, las comunicaciones alemanas comenzaron a ser descifradas totalmente, un hecho que ha quedado en la historia del criptoanálisis. Y la situación siguió así durante varios años, aunque la única complicación residía en las comunicaciones de la Marina pues ellos ya usaban cinco rotores de los cuales elegían tres para las transmisiones.

Con cada cambio que los alemanes hacían, el catálogo de posibilidades perdía sentido. Por ese motivo Rejewski diseñó una máquina, bautizada "bomba", consistente en seis pares de Enigmas que trabajaban en paralelo para explorar las 17.576 posibilidades. El diseño y los planos fueron suministrados a la empresa AVA –la misma que había comprado una Enigma comercial–, la cual fue capaz de entregar lo solicitado en noviembre de 1938. Se usaban seis "bombas" porque la velocidad era esencial debido a que los alemanes cambiaban sus claves a la medianoche, y entonces trabajar en paralelo reducía sensiblemente a dos horas el tiempo requerido. Como dato curioso vale la pena mencionar que el mayor Gwido Langer, jefe del Biuro Szyfrów, seguía recibiendo claves del espía Schmidt que se usaban diariamente, con lo cual el trabajo de los criptoanalistas podría haberse considerado inútil. Pero Langer creía indispensable que siguiesen trabajando continuamente pues creía que algún día la conexión con Schmidt podría terminarse, y el tiempo le dio la razón.

A fines de 1938 la situación comenzó a complicarse. Los alemanes empezaron a usar cinco rotores distintos, tal como

venía haciendo la Marina, de los cuales elegían tres, lo que implicaba diez combinaciones posibles que, multiplicadas por 6 según la disposición de los tres elegidos, daba como resultado 60. O sea que la complejidad había sido aumentada por un factor 10. Por ello, necesitaba multiplicar por 10 el número de "bombas" pero el costo de construirlas era del orden de quince veces el presupuesto del Biuro Szyfrów. El panorama se oscureció aún más cuando los alemanes aumentaron el número de cables de seis a diez, lo que producía una máxima complejidad, como vimos anteriormente.

Aparte de las "bombas", Henryk Zygalski había inventado un método manual consistente en hojas de papel con unas mil perforaciones hechas con una estructura determinada. Mediante veintiséis hojas, cada una de las cuales representaba una posible posición de un rotor, que se apilaban una después de otra, si aparecía un agujero común a todas ellas, eso indicaba el posicionamiento secreto de la Enigma. Pero su uso dejó de ser efectivo cuando los alemanes introdujeron los cambios en las Enigma pues también habría que haber fabricado sesenta conjuntos de hojas perforadas.

Tanto las hojas perforadas como las "bombas" eran métodos ingeniosos de lidiar con una gran cantidad de permutaciones.

A principios de 1939 la situación era muy preocupante porque los criptoanalistas ya no podían descifrar los mensajes y los contactos con Schmidt se habían cortado precisamente cuando disponer de las claves era vital dada la proximidad de la guerra. La idea de la *blitzkrieg* (guerra relámpago) de Hitler se basaba en ataques muy coordinados entre las divisiones de tanques, la infantería y la fuerza aérea, y su *leitmotiv* era "velocidad de ataque mediante velocidad de las comunicaciones". En definitiva, si los polacos no

podían seguir descifrando las comunicaciones alemanas, era inminente que se produjera un desastre militar.

Langer, como jefe del Biuro Szyfrów, tenía claro que, si Polonia era invadida, su deber era transferir todos los brillantes resultados de sus criptoanalistas a los Aliados dado que muy posiblemente Francia e Inglaterra, países que disponían de muchos más recursos técnicos y económicos, estarían en condiciones de fabricar las "bombas" que se necesitaban. Por ese motivo, el 24 de julio de 1939 organizó una reunión con los mejores criptoanalistas franceses e ingleses en Pyry, un pueblo donde los polacos habían construido una estación de codificación y transmisión, en la cual los asombró con una de las "bombas" construidas por Rejewski. También en esa reunión se enteraron de que los polacos venían descifrando hace tiempo las comunicaciones alemanas, lo cual fue un gran impacto para los franceses puesto que los logros se habían basado en los datos suministrados por el espionaje de su propio país, algo esencial que habían descartado. Finalmente Langer les dio dos réplicas de la Enigma, así como los planos de las "bombas". Todo ese material se despachó por correo diplomático a París y, de allí, una de las Enigmas fue llevada a la estación Victoria de Londres por el capitán Bertrand en persona. Dos semanas después Alemania invadió Polonia y comenzó la guerra.

Los ingleses no habían tenido una relación con los expertos polacos como la de los franceses, y por eso mantuvieron un contacto distante hasta la conferencia de Pyry. Pero sí tenían expertos criptoanalistas. El más destacado de ellos era Alfred Dillwyn Knox, educado en Eton y Cambridge, quien había sido reclutado por la Marina inglesa durante la Primera Guerra Mundial e incorporado en la famosa *Room*

40, donde había comenzado el trabajo criptoanalítico. Knox era famoso por haber descubierto el código que los alemanes usaban para comunicarse con sus submarinos.

A mediados de la década de 1930, la *Room 40* había sido reorganizada: se convirtió en la Escuela de Cifrados y Códigos, dirigida por el comandante Alastair Denniston, en la que Knox trabajó junto con otros destacados expertos, como Nigel de Grey, quien había trabajado en el problema del telegrama Zimmermann. Knox también había conseguido una Enigma comercial y había utilizado los datos obtenidos por el espía Schmidt, pero no pudo lograr los extraordinarios resultados de los polacos, aunque estuvo cerca de resolver el problema.

Denniston había reclutado a una cantidad de mentes brillantes, entre las que se contaban Dennis Babbage, pariente lejano de Charles Babbage, y Alan Turing. Su "escuela" se había mudado a Bletchley Park, un parque que tenía una gran mansión que estaba situado entre Oxford y Cambridge, rodeado por alambrados de púas y custodiado por la Real Fuerza Aérea [Royal Air Force; RAF]. Inicialmente las instalaciones albergaban a unas doscientas personas, pero llegó a tener aproximadamente diez mil a medida que las operaciones se tornaban más complejas: había un sector dedicado a las comunicaciones de la *Wehrmacht*, otro a la *Kriegsmarine*, etc. De hecho, coexistían diversas variaciones de las máquinas Enigma, e incluso había distintos niveles de seguridad, lo que implicaba que se necesitaban diferentes equipos de especialistas.

El principal problema que enfrentaban en Bletchley Park no era que se interceptaran gran cantidad de comunicaciones en distintos lenguajes de diferentes fuentes y cifrados por diversas Enigmas, sino cómo procesar en forma útil todo

lo que se descifraba. Para proteger las actividades del espionaje alemán había que ocultar los orígenes de la información pero, al mismo tiempo, las fuerzas armadas se resistían a usar datos cuyo origen no podían controlar. Lentamente el Ejército y la RAF accedieron a centralizar operaciones en Bletchley Park, pero la Marina quería hacer todo a través de su propio Centro de Operaciones y exigía que les enviaran los mensajes en su idioma original. Finalmente accedieron a integrarse en julio de 1940.

A fin de transmitir solo los mensajes realmente importantes que habían sido descifrados, formaron una unidad especial llamada Ultra y comenzaron a usar el anotador de única vez para dar seguridad total y evitar el espionaje alemán. Sin embargo, la Marina no quiso usar este método y continuó empleando sus propios códigos.

Alemania había establecido una gran organización dedicada al cifrado y al criptoanálisis, pero seguía convencida de que la Enigma era inexpugnable aunque, de hecho, los mensajes de sus fuerzas armadas eran descifrados diariamente por los Aliados. Si bien en diversas ocasiones se dieron cuenta de que algunos de sus mensajes habían caído en manos enemigas, siempre lo atribuyeron a que habían sido suministrados por espías. Por ejemplo, al invadir Francia, diversos archivos incautados en París mostraban textos en alemán que habían sido codificados mediante Enigmas y que habían sido obviamente leídos por los franceses. Luego de investigarlos llegaron a la conclusión de que todo se debía al trabajo de un espía prusiano, quien supuestamente les había vendido a los franceses las claves de un cierto período.

Por su parte, los códigos ingleses venían de un largo período de debilidades pues habían sido quebrados por los

alemanes. El Almirantazgo tenía conciencia de la situación, así que cuando estalló la guerra procedieron a complicar sus métodos y a asignar distintas variaciones de estos a buques diversos. Con esas medidas el trabajo criptoanalítico de los alemanes se complicó, y aunque lograban descifrar las señales interceptadas, lo hacían tan lentamente que no obtenían ventajas operacionales. Sin embargo, ello les permitió entender las estrategias navales de los ingleses y tener también información sobre muchas unidades navales. Por su parte, los expertos de Bletchley Park no podían descifrar las comunicaciones alemanas. Para cambiar esa situación, la Marina inglesa procedió a complejizar sus métodos de codificación, lo que condujo a que el enemigo dejase de ubicar sus barcos como lo venía haciendo. Pero los ingleses cometieron un error análogo al de los alemanes: manejar también métodos mucho más simples de cifrado en una parte de sus comunicaciones, en particular la de los convoyes que llevaban suministros. De este modo los submarinos sabían dónde esperar a los barcos mercantes y, por eso, las pérdidas llegaron hasta el punto de que hundían un barco cada dieciséis horas. En ese momento los alemanes tomaron una decisión razonable desde la perspectiva táctica pero que comprometía su seguridad: en vez de atacar mediante submarinos solitarios empezaron a hacerlo en flotillas, y eso condujo a que tuviesen que incrementar drásticamente el volumen de las comunicaciones radiales entre ellos, exponiéndose así a ser interceptados por los Aliados.

La Marina alemana seguía creyendo que las Enigma eran inviolables, pero al mismo tiempo sabían que la capacidad criptoanalítica de los ingleses era notable. Por eso comenzaron a cambiar los rotores con mucha más frecuen-

cia. También limitaron la longitud de los mensajes codificados con la misma clave y comenzaron a usar hasta siete rotores.

Otra medida de precaución fue la de establecer diferentes redes, cada una de las cuales servía para un cierto propósito y tenía sus propias claves. Las más conocidas fueron Hydra, Aegir, Alpha, etc. Otra, llamada Neptuno, se reservó para la flota principal y misiones especiales y, dado que se usaba muy poco, fue muy difícil de quebrar. Finalmente el quiebre de Hydra facilitó el descifrado de Neptuno en la primavera de 1941.

Para el esfuerzo criptoanalítico era esencial identificar en forma correcta a qué red pertenecía cada mensaje pues cada una tenía sus códigos y modalidades operativas. Nuevamente los alemanes no se dieron cuenta de que cada operador humano tiene características propias, como la cadencia al escribir, etc. El tema era tan repetitivo que, cuando los expertos de Bletchley Park escuchaban una transmisión en Morse, enseguida identificaban la red que la originaba.

En 1942 apareció una nueva red de extrema importancia llamada Tritón, usada por el almirante Dönitz, comandante de los cien submarinos que estaban en operaciones. Los criptoanalistas encontraron muchas barreras para decodificar sus mensajes, tarea que les llevó más de un año, durante el cual, las pérdidas navales de los Aliados fueron considerables. Pero al descifrarla pudieron determinar las ubicaciones de los submarinos. Otras redes con propósitos específicos nunca fueron quebradas.

Un cambio impactante tuvo lugar en 1943 cuando los alemanes introdujeron una nueva Enigma con cuatro rotores operacionales. Esta máquina causó muchos problemas

logísticos pues era necesario cambiar todas las existentes en barcos y submarinos, cosa que finalmente se logró: desde la medianoche del 8 de marzo de 1943 todas las comunicaciones comenzaron a ser encriptadas con el nuevo modelo. Los Aliados dejaron de tener acceso a los mensajes, justamente cuando los preparativos para el día D habían comenzado y, por ende, en el momento en que la Batalla del Atlántico era crucial dado que los convoyes solamente podían ser protegidos por la aviación hasta 200 millas de las costas de los Estados Unidos o Inglaterra. Fuera de esas zonas los barcos eran presa fácil para los submarinos.

Como vimos anteriormente los matemáticos polacos diseñaron la "bomba", que fue totalmente eficaz para descifrar las primeras máquinas Enigma. Pero cuando el sistema utilizado se tornó más complejo era necesario multiplicar la capacidad operativa por un factor 10, algo imposible con los escasos medios disponibles en Polonia. Todos los resultados obtenidos por Rejewski, Jerzy Różycki y Henryk Zygalski fueron transferidos a los Aliados y, en particular, a Bletchley Park, donde Alan Mathison Turing, genial matemático y lógico británico nacido el 23 de junio de 1912, no tuvo dificultades para entender el sistema de las "bombas" polacas, ni la propuesta para construir su versión más compleja destinada a descifrar los códigos de la Enigma Naval.

Como es sabido, Alan Turing hizo enormes contribuciones a la ciencia de la computación: por ejemplo, la máquina universal y el test que llevan su nombre fueron fundamentales para intentar determinar el posible comportamiento "inteligente" de una computadora. En particular, el test fue el precursor de la inteligencia artificial.

Luego de la guerra Turing se incorporó a la Universidad de Manchester, donde partició del proyecto de la computadora Ferranti Mark I al diseñar parte de su software con la ayuda de la doctora Cicely Popplewell, quien viajó años después a Argentina para dictar el primer curso de programación –del cual el autor de este libro fue alumno– en la Facultad de Ciencias Exactas y Naturales de la UBA, que había adquirido una computadora de esa marca.

En 1952 fue procesado por su homosexualidad debido a que en esa época, y tal como había sucedido con Oscar Wilde medio siglo antes, aún se consideraba un delito. Falleció el 7 de junio de 1954, aparentemente al suicidarse con cianuro luego de haberse sometido a una castración química que le había traído serios problemas físicos.

Como alguien dijo una vez, si las autoridades hubiesen sabido de su homosexualidad, muy probablemente los Aliados habrían perdido la guerra. Alan Turing recién fue reivindicado en 2013 por la reina Isabel.

Otro aspecto interesante de la historia lo constituyen las debilidades de las comunicaciones de la *Luftwaffe* que fueron sistemáticamente descifradas por los ingleses. El 17 de julio de 1940 los expertos descifraron un mensaje del mariscal Hermann Göring a sus comandantes para comenzar la operación Sea Lion, destinada a invadir las islas británicas, dado que el gobierno inglés no mostraba señales de aceptar la propuesta de Hitler para dividirse el mundo entre ambos.

Las situaciones de las respectivas fuerzas aéreas eran las siguientes: Inglaterra ya tenía la capacidad de fabricar muchos más aviones que Alemania, pero disponía de muchos menos pilotos. Por ese motivo era vital que usara

eficazmente su aviación pues podían llegar a perder la batalla de Inglaterra por la carencia de recursos humanos calificados. Y, nuevamente, la necesidad imperiosa de ocultar el hecho de que Bletchley Park descifraba los mensajes de la *Luftwaffe* condujo a serios problemas internos, pues esa información solo era conocida por el mariscal Hugh Dowding, jefe de la fuerza aérea británica, quien tomó medidas destinadas a desorientar al espionaje alemán que, al no ser ni entendidas ni compartidas por sus oficiales, llevaron a que pidiesen su retiro.

El punto culminante fue el 15 de agosto, cuando Göring lanzó al ataque sus tres flotas, incluyendo por primera vez la que estaba en Dinamarca y Noruega, con el propósito de sorprender a los ingleses y sin saber que sus comunicaciones ya descifradas habían sido transmitidas a Churchill a través de la red Ultra. El mariscal Dowding condujo la defensa usando la información de los mensajes de la *Luftwaffe* y el análisis de sus tácticas, y produjo enormes pérdidas al enemigo. A pesar de ello, en septiembre la *Luftwaffe* volvió a atacar con 328 bombarderos y 769 cazas que fueron sorpresivamente enfrentados por 300 Spitfires y Hurricanes sobre la base de la información interceptada. Las pérdidas alemanas fueron muy grandes comparadas con las inglesas, con el resultado final de que el 17 de septiembre los técnicos de Bletchley Park descifraron un mensaje que decía que Hitler había dado la orden de abandonar los ataques a las islas.

Una dificultad para descifrar las señales de la Marina provenía del hecho de que todos los esfuerzos de los criptoanalistas polacos se habían centrado en el quiebre de las comunicaciones del Ejército alemán pues lo que temían era

la invasión terrestre. Como vimos, las codificaciones de la Marina eran las más complejas. A su dificultad intrínseca se sumaba entonces el hecho de que no les habían prestado atención. Por ello siempre se había intentado capturar equipos, algo que se logró a fines de 1939 cuando un submarino dañado que viajaba a la deriva y con su tripulación muerta fue abordado por los ingleses, quienes encontraron una Enigma con siete rotores (tres de los cuales aún estaban operando). Ese era un hecho excepcional pues todas las unidades tenían órdenes estrictas de destruir por completo el equipamiento y la documentación. Al respecto es interesante ver <enigmamuseum.com> donde se describen expediciones realizadas en Europa para recuperar las Enigma descartadas, pues diversas unidades militares alemanas no procedieron a destruir el equipamiento de acuerdo con las directivas recibidas sino que simplemente las arrojaron en aguas profundas.

La imposibilidad de descifrar los códigos de la Enigma M4 de cuatro rotores operativos sumió en la oscuridad los esfuerzos de inteligencia aliados. Así es como toda la campaña de Noruega se realizó sin ninguna información proveniente de las comunicaciones navales, excepto las mediciones del volumen de tráfico, indicativas de que hechos importantes estaban ocurriendo, y la debilidad sistemática de los códigos de la *Luftwaffe,* que permitían anticipar los ataques aéreos. Se conseguía información naval importante cuando la *Luftwaffe* hacía operaciones conjuntas dado que su tráfico sí se descifraba. Aquí hay que aclarar que la M4 se denominaba así porque se agregaba un rotor más (llamado *Zusatzwalze*) a la izquierda de los tres movibles, que intervenía en el cifrado, pero que permanecía fijo.

Es interesante notar que parte de las debilidades provenían de los errores operativos pues, por ejemplo, en vez de decidir al azar el código que se iba a utilizar a partir de cada medianoche, los operadores de la *Luftwaffe* a menudo elegían la posición final en la que habían quedado los rotores luego de la última encriptación del día. En cambio la *Kriegsmarine* cambiaba los rotores con gran frecuencia, elegía las claves al azar, etc., y, por sobre todas las cosas, utilizaba la totalidad de los medios disponibles de comunicaciones terrestres, minimizando el uso las radios.

Un gran éxito ocurrió el 9 de mayo de 1941, cuando el submarino *U-110* atacó un convoy pero fue descubierto por la corbeta *Aubretia*. Se sumergió inmediatamente, pero fue alcanzado por cargas de profundidad y tuvo que emerger. Al hacerlo, fue atacado por los barcos ingleses. El capitán era Fritz-Julius Lemp, tristemente famoso por haber hundido el barco de pasajeros *Athenia* el día que la guerra comenzó, el 3 de septiembre de 1939, a pesar de que la convención de La Haya lo prohibía. En esa fecha comandaba el submarino *U-30*, y ciento dieciocho personas murieron en el hundimiento del barco. Lemp se arrojó al agua y al ver que el submarino no se hundía trató de regresar a bordo para destruir la Enigma, pero fue herido por una bala de ametralladora. Los ingleses pudieron capturar la máquina con sus rotores, los libros de claves y manuales operativos. Dado que las patrullas submarinas duraban entre dos y tres meses, los libros de claves abarcaban esos períodos, así que hasta el mes de junio Bletchley Park pudo descifrar sin problema todas las comunicaciones realizadas con la máquina Enigma M.

Es interesante notar que en esta batalla continua entre codificadores y decodificadores hubo muchos errores que

facilitaron el quiebre de las comunicaciones. Por ejemplo, los alemanes construyeron dos buques de guerra gigantes: el *Bismarck* y el *Tirpitz*. Los informes de inteligencia crearon una gran preocupación para los Aliados dada la importancia vital que tenía la Batalla del Atlántico. Los ingleses, aunque contaban con información fragmentada –pues no todas las comunicaciones habían sido descifradas– y pocas posibilidades de reconocimiento aéreo –debido al mal tiempo–, enviaron para interceptar al *Bismarck* el crucero *Hood*, el *Prince of Wales*, el *Suffolk* y seis destructores. Los barcos se encontraron por casualidad, el *Hood* fue hundido, el *Prince of Wales* dañado y el *Bismarck* comenzó a perder combustible. Los ingleses, anticipando que los alemanes se iban a dirigir al Atlántico, hicieron zarpar al portaaviones *Ark Royal*, al crucero *Sheffield* y al *Renown*, que estaban en Gibraltar.

La Inteligencia española, que cooperaba con los alemanes, avisó del movimiento de los barcos, pero la gente de Bletchley Park descifró sus mensajes unos días más tarde, aunque eso solo les sirvió para enterarse de que los alemanes estaban sobre aviso. Sin embargo, afortunadamente para los Aliados, la *Luftwaffe* seguía convencida de la invulnerabilidad de sus Enigma, pero al descifrar sus comunicaciones referentes a la cobertura aérea del *Bismarck* dedujeron que se dirigía a Francia. Peor aún, un oficial de alto rango de la *Luftwaffe* estacionado en Atenas que tenía un hijo sirviendo en el *Bismarck* preguntó por el destino del barco y le respondieron que se hallaba en Brest usando los códigos de la diplomacia, que eran muy fáciles de romper. O sea, dos errores cruciales: el primero fue darle información esencial a una persona que no tenía por qué tenerla

y el segundo, utilizar códigos débiles. En consecuencia, un avión Swordfish del *Ark Royal* disparó un torpedo que inutilizó el timón del *Bismarck* y, a partir de ese momento, los barcos ingleses *Rodney* y *Dorsetshire* terminaron de rematarlo.

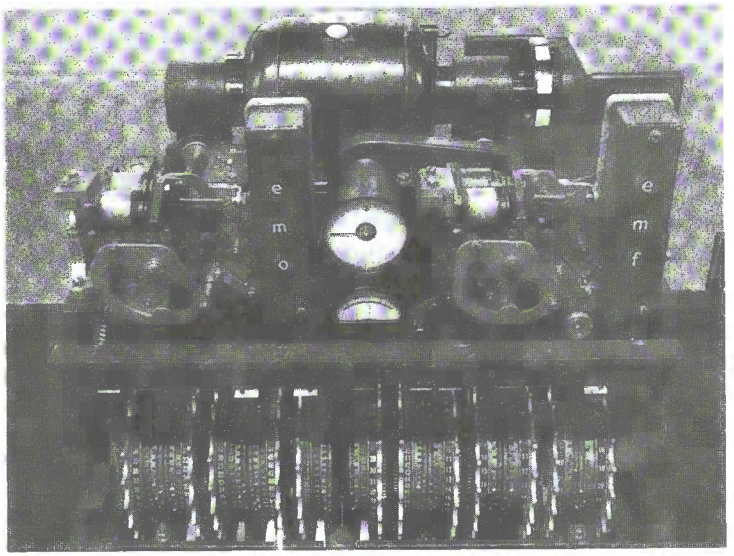
La estrategia alemana de usar barcos de gran porte había sido un fracaso, y esto se debió en gran medida a sus errores de comunicaciones. A partir de ese momento la suerte de la Batalla del Atlántico recayó en el almirante Dönitz y sus submarinos.

CÓMO SE QUEBRÓ LA MÁQUINA LORENZ

A pesar del convencimiento de los alemanes, fomentado por sus "expertos", Hitler y su Alto Mando se comunicaban mediante otra máquina, la Lorenz SZ-40/42, dedicada a enviar señales de télex. Generaban grupos de cinco bits seudoaleatorios que eran combinados con los del texto que se iba a encriptar mediante la operación XOR que vimos anteriormente al tratar el anotador de única vez. Tenía doce rotores y el cifrado resultante era mucho más complicado que el de las máquinas Enigma. El número de combinaciones posibles de claves era de $1{,}05 \times 10^{170}$, gigantesco aun para las computadoras actuales. Para intentar quebrarla era necesario pensar en un enfoque distinto al utilizado con las Enigma.

Todas las máquinas cifradoras adosadas a los equipos de télex fueron bautizadas Fish (pescado) por la gente de Bletchley Park; a la Lorenz en particular se la conocía por Tunny (atún).

Figura 2.4
Máquina Lorenz



Fotografía: cortesía del Enigma Museum (enigmamuseum.com).

En 1940 los ingleses interceptaron señales cifradas que no se correspondían con el código Morse, y de hecho era el código Baudot de télex. El 30 de agosto de 1941 un operador de códigos de Viena envió un texto de unos 4000 caracteres a un colega en Atenas. El receptor le dijo que no había entendido todo y solicitó su retransmisión, algo que hizo la persona de Viena utilizando la clave original y empleando una cierta cantidad de abreviaturas. Eso permitió que en un lapso inusualmente breve de tiempo un equipo liderado por el coronel John Tiltman descifrara el mensaje. Todo el procedimiento fue enviado

a un joven matemático de Cambridge llamado Bill Tutte, quien en solo cuatro meses consiguió construir una máquina electromecánica análoga, la Tunny. Esta máquina facilitó el análisis manual de los mensajes cifrados, pero el proceso era sumamente lento, como era de esperar, y por ese motivo se encaró el diseño y construcción de la primera computadora electrónica, llamada Colossus. Esta fue diseñada por Max Newman tomando en cuenta diversos resultados conseguidos por Alan Turing acerca de los sistemas Fish, y construida por un equipo liderado por el ingeniero electrónico Tommy Flowers. Esta computadora se utilizó para quebrar los códigos producidos por la máquina Lorenz y no para descifrar los generados por la Enigma.[4]

Es importante destacar que, según la historia aceptada, la ENIAC [*Electronic Numerical Integrator And Computer*; computador e integrador numérico electrónico] fue la primera computadora electrónica terminada de construir en 1946.

Anteriormente existió la calculadora mecánica Z1, inventada por Konrad Zuse en Alemania en 1936, que era eléctrica, usaba el sistema binario y se podía programar en forma muy limitada mediante instrucciones en cintas perforadas. Hubo dos modelos posteriores, las Z2 y Z3, pero todas presentaron problemas serios para funcionar correctamente.

Hoy en día la Colossus es considerada la primera computadora totalmente digital y electrónica, aunque su existencia se mantuvo en secreto hasta 1974 por razones militares.

Al poco tiempo de terminada la Segunda Guerra Mundial, se ordenó destruir las máquinas Colossus, excepto dos, que se conservaron para la Guerra Fría, pero con el advenimien-

4. Véase "Colossus: Birth of the Digital Computer", en <www.cryptomuseum.com>.

to de nuevas tecnologías ellas también fueron desmanteladas. En 1991 un equipo liderado por Tony Sale, fundador y primer curador del museo de Bletchley Park, encaró el proyecto de reconstruir una máquina Colossus, tarea que afortunadamente se completó en 2007.

Inglaterra había capturado miles de máquinas Enigma que fueron distribuidas a sus ex colonias, donde, como en su momento los alemanes, creían que sus códigos eran invulnerables. Esto les permitió a los ingleses descifrar fácilmente las comunicaciones, con los consiguientes beneficios políticos y económicos que eso supone. Por ello era importante conservar el secreto de lo que se había logrado en Bletchley Park.

¿QUÉ SUCEDÍA EN LA GUERRA DEL PACÍFICO?

La criptología estadounidense tomó impulso en 1917 cuando el país entró en la Primera Guerra Mundial. Se creó una oficina especializada, denominada la "cámara negra", a cargo de Herbert Yardley, que comenzó a interceptar comunicaciones japonesas y tuvo su primer éxito en 1920 al quebrar el código diplomático.

Los métodos de encriptación utilizados por Japón eran bastante simples y, probablemente, habrían continuado sin cambios si no hubiera sido por una terrible equivocación de los estadounidenses. Lo que sucedió fue que el Ejército decidió formar un Servicio de Señales de Inteligencia, a cargo de un criptólogo talentoso llamado William Friedman, y el jefe del Estado Mayor decidió unir ambas organizaciones bajo el mando de Yardley, pero con un salario mucho menor del que tenía. Yardley no aceptó y decidió publicar en *The Saturday*

Evening Post tres extractos de un libro que estaba por sacar, titulado *The American Black Chamber*, en los que describía el quiebre de los códigos diplomáticos nipones. Esa tremenda filtración condujo a que los japoneses dejasen de lado los métodos que venían usando y recomenzaran desde cero su criptografía.

Como el idioma japonés consta de miles de caracteres *kanji*, es totalmente inadecuado para la criptografía y por eso utilizaron el alfabeto latino. Una vez decidido esto, se interesaron naturalmente por las máquinas Enigma; compraron primero su versión comercial y, posteriormente, la adoptaron tanto para sus Fuerzas Armadas como para el gobierno. Pero los japoneses querían desarrollar una máquina propia diferente de la Enigma pues sus aliados alemanes podrían constituirse en potenciales enemigos futuros. Para ello consiguieron detalles constructivos de diversas máquinas de rotores, como la diseñada por Friedman, llamada Sigaba, la británica Typex y la sueca de Boris Hagelin.

La idea de Friedman era muy interesante pues modificaba el funcionamiento normal de las Enigma, las cuales, al codificar, avanzaban una posición de los rotores. En cambio Friedman agregó una cinta perforada de modo tal que, cuando se presionaba una tecla con una letra dada, la máquina la encriptaba como la Enigma pero a la vez la corriente eléctrica pasaba a través de la siguiente perforación de la cinta y definía un movimiento no secuencial de los rotores. Sin embargo, el uso de cintas de papel no era en absoluto práctico en condiciones de combate, así que un joven contratado por Friedman, Frank Byron Rowlett, propuso tener un juego extra de rotores para generar los movimientos al azar que necesitaban los que realmente encriptaban. Ese modelo de

la Sigaba fue bautizado M-134-C, y hasta el día de hoy no se sabe si alguien lo pudo criptoanalizar.

Por otra parte, los japoneses le prestaron especial atención a las 140.000 máquinas Hagelin M-209 compradas por el Ejército, pero los intentos de espionaje fracasaron y tampoco pudieron capturarlas en el campo de batalla porque recién se enfrentaron con las tropas estadounidenses en las postrimerías de la guerra. Debido a esto decidieron acudir a las fuentes y compraron varias máquinas en Estocolmo a través de su embajada, que luego trasladaron con submarinos desde el mar Báltico hasta Japón.

La máquina que finalmente construyeron sobre la base de toda la información recibida incluía naturalmente diversas modificaciones y la llamaron Purple (púrpura).

Los analistas estadounidenses que intentaban descifrar el código Purple enfrentaron problemas similares a los que habían tenido los polacos con las Enigma, pero no disponían de ningún espía como Asché y tampoco tenían posibilidades de conseguirlo debido a las impenetrables estructuras japonesas. Lo ideal era conseguir largos textos encriptados con la misma clave para poder arribar a conclusiones. Años después, cuando le preguntaron a Rowlett cuál había sido su mayor contribución al esfuerzo bélico dijo: "Mi convicción de que el código Purple se podría quebrar".

En esa época los Estados Unidos no compartían los grandes avances en criptología iniciados por los matemáticos polacos y continuados en Bletchley Park porque eran neutrales. La situación cambió con la derrota de Francia y la amenaza de Hitler de invadir Inglaterra pues entonces Churchill se aproximó a los Estados Unidos en busca de armamento y ayuda en general. Eso condujo a un acuerdo para

intercambiar los secretos tecnológicos más preciados, como los logros en el descifrado de códigos, el radar, los sistemas de detección de submarinos, etc. También los ingleses recibieron una de las primeras máquinas Purple reconstruidas por los estadounidenses, un paso esencial para analizar los códigos. Asimismo organizaron una sección de Bletchley Park dedicada a los códigos japoneses.

Un enorme éxito del trabajo de los criptoanalistas ocurrió en 1942 cuando se logró descifrar un mensaje japonés en el cual se decía que el almirante Isoroku Yamamoto, comandante en jefe de la Flota Combinada de la Armada Imperial Japonesa, iba a atacar un objetivo militar estadounidense cuyo código era "AF". En ese momento la superioridad naval japonesa era enorme y la única manera de contrarrestarla era aplicar un golpe decisivo basado en información de inteligencia.

Yamamoto era un estratega muy destacado, había liderado el ataque a Pearl Harbor y su objetivo en ese momento era destruir los portaviones estadounidenses *Enterprise*, *Hornet* y *Yorktown*, que eran los únicos en condiciones de operar (apenas un mes antes, en la Batalla del Mar de Coral, el *Yorktown* había sido dañado y el *Lexington* hundido).

Para engañar a los japoneses diseñaron una trampa que resultó efectiva. La Inteligencia aliada usaba varios códigos y sabía que uno de ellos había sido quebrado por los japoneses. Mediante ese código transmitieron un mensaje que decía que la isla de Midway se estaba quedando sin agua potable por problemas con el sistema de desalinización. Al poco tiempo los japoneses, usando su código JN-25, transmitieron que "AF" tenía problemas de agua potable. Entonces estaba claro que el ataque secreto a Midway había quedado al descubierto y la consiguiente batalla, que fue considerada

la más importante del Pacífico, les ocasionó grandes pérdidas a los japoneses. Concretamente perdieron cuatro portaviones, un crucero pesado y doscientos cuarenta aviones.

El siguiente golpe mortal sucedió cuando en abril de 1943 la Inteligencia aliada descifró un mensaje que indicaba que el almirante Yamamoto iba a hacer un viaje de inspección a las islas Salomón, y especificaba el día y la hora. Debido a que el almirante era conocido por su puntualidad, el avión Mitsubishi G4M en el que volaba fue interceptado y derribado por una escuadra de aviones estadounidenses P-38. Su muerte fue determinante para minar la moral de los japoneses.

Como conclusión se puede afirmar que la Batalla del Pacífico se ganó gracias a que se descifraron los códigos japoneses.

¿QUÉ OCURRÍA EN OTROS ESCENARIOS EUROPEOS?

Como ya mencionamos, la organización Ultra era la encargada de coordinar todos los datos de inteligencia. Otro de sus logros fue determinar los movimientos de los alemanes en el Mediterráneo, lo que permitió destruir las líneas de abastecimiento del mariscal Erwin Rommel, prever la invasión de Grecia y preparar la invasión aliada en Sicilia.

Ultra también fue concluyente en la preparación del Día D al permitir ubicar con gran precisión las posiciones de las tropas alemanas.

Un aspecto esencial en el descifrado de mensajes es conocer el idioma en el que fueron escritos. Por eso recurrir a los indios navajos para transmitir mensajes en su lengua

fue una medida muy inteligente por parte de los Estados Unidos, aunque trajo aparejada la necesidad imperiosa de impedir que alguno fuese capturado por el enemigo, ya que en ese caso la orden era matarlo.

Es verdad que la existencia del espía Asché fue muy importante para el extraordinario trabajo de los criptoanalistas polacos, pero se considera que sin su información el quiebre de la Enigma igual se habría logrado, aunque habría demorado un par de años más.

Los códigos alemanes y japoneses fueron quebrados sobre todo gracias a la cantidad de errores cometidos. En primer lugar, el uso de códigos de distinto grado de fortaleza, olvidando que el eslabón más débil es el que determina la resistencia de una cadena. Así, vimos que la *Luftwaffe* tenía una creencia ciega en la superioridad de sus métodos, pero sus mensajes se descifraban sin problema. Y la *Kriegsmarine*, que tenía procedimientos mucho más seguros, debilitó fuertemente sus comunicaciones al interactuar con el débil eslabón de la *Luftwaffe*.

También cometieron errores de protocolo, como elegir la clave para el día siguiente según las tres últimas letras de los rotores del día anterior o el encriptar sucesivamente las letras del nuevo código, dado que la segunda terna era la consecuencia del avance de los rotores. Sin embargo, el punto que resultó esencial fue la incapacidad de las Enigma de encriptar una letra en sí misma.

Las máquinas de rotores Typex inglesa y la estadounidense Sigaba eran más complejas que las Enigma, pero sus códigos nunca pudieron ser quebrados, en gran medida gracias a la rigurosidad en las operaciones de los Aliados, que evitaron los errores de los alemanes.

¿Cómo llega la encriptación al mundo civil?

Acerca de Lucifer, de la televisión codificada y de cómo administrar claves cada vez más numerosas

Tradicionalmente la criptografía estaba restringida a los ámbitos militares hasta que en 1974 la empresa IBM, que había contratado a un criptógrafo alemán llamado Horst Feistel, quien desarrolló un método de cifrado llamado Lucifer, se lo presentó al gobierno estadounidense solicitando permiso para usarlo en aplicaciones civiles. La agencia que hoy en día se conoce como Instituto Nacional de Estándares y Tecnología [National Institute for Standards and Technology; NIST], en consulta con la Agencia de Seguridad Nacional [National Security Agency; NSA], estudiaron el método, lo modificaron acortando la longitud de sus claves a 56 bits y lo denominaron DES [*Data Encryption Standard*; estándar de cifrado de datos].

En este punto, la historia es controvertida pues la comunidad criptográfica siempre estuvo convencida de que los cambios en el diseño del algoritmo, así como el acortamiento de las claves, tenían como objetivo que el gobierno fuese capaz de descifrar los mensajes codificados con el DES. Por otra parte, en esa época se creía que la encriptación solo

podía realizarse por hardware, y como IBM dominaba el mercado mundial, bastaba con controlar esa empresa para hacerlo.

Este método es el más estudiado del mundo desde su publicación como estándar el 17 de marzo de 1975 en el Registro Federal de la Oficina Nacional de los Estados Unidos para aplicaciones "no clasificadas". Fue quebrado definitivamente en 1998 por medio de una computadora llamada DES *cracker*, desarrollada por la Electronic Frontier Foundation, que logró hacerlo en cincuenta y seis horas. Al año siguiente esta computadora, junto con un sistema distribuido, consiguió bajar el tiempo a veintidós horas y quince minutos.

Como vimos al tratar los monoalfabéticos, si un método conserva las propiedades estadísticas de los lenguajes, como la frecuencia de aparición de las letras, digramas y trigramas, entonces es fácilmente descifrable. Feistel desarrolló las llamadas "cajas" que llevan su nombre, para lograr lo que en criptografía se denomina "confusión y difusión", destinadas a que un texto cifrado aparezca como un conjunto de bits aleatorios.

Como ejemplo, supongamos que tenemos un texto de varios miles de letras "a". Si lo ciframos con un método de sustitución elemental que transforme la letra "a" en "h", el resultado será un texto con tantas "h" como "a" tenía el original. En cambio, un buen método debe dar como resultado algo como "v/&fw?...", sin ninguna estructura detectable, y en el cual todos los símbolos posibles de una computadora aparezcan aproximadamente con la misma frecuencia. Esta propiedad deseable también sirve para disimular el texto relevante. Por ejemplo, si un espía quiere mandar el mensaje "la cita es

mañana a las 10 hs", primero le agrega una gran cantidad de espacios que serán transformados en un galimatías inentendible, pero que al ser descifrados con la clave correcta volverán a aparecer como espacios.

Otra cualidad deseable es conocida como "efecto avalancha", y consiste en que, si se cambia un bit del mensaje o de la clave, el texto cifrado difiera del original en al menos la mitad de los bits. La idea intuitiva es que si un *hacker* está probando distintas claves, aunque una de ellas sea prácticamente igual a la verdadera, el resultado siga siendo ininteligible y no logre ninguna proximidad con el objetivo buscado.

Los conceptos de confusión y difusión fueron estudiados por Claude Shannon al desarrollar la teoría de la información.

El DES fue muy exitoso en los sistemas bancarios y financieros, y aún hoy los ATM (*automatical transaction machine*; cajeros automáticos) los utilizan. Las claves de un cajero son números llamados PIN [*Personal Identification Number*; número de identificación personal], y la Asociación de Banqueros de los Estados Unidos estableció en 1982 un estándar para administrarlos y darles seguridad basándose en el DES. Análogamente se desarrollaron normas para la autenticación de mensajes, la distribución de claves, etc.

¿QUEBRAR CÓDIGOS USANDO LA "FUERZA BRUTA"?

La seguridad de un método de encriptación depende de dos factores: su robustez matemática y la longitud de las claves que emplea. Si el método no es robusto podrá ser quebrado aunque utilice claves de gran longitud.

Un ejemplo de esto es el uso de permutaciones del alfabeto que vimos anteriormente, pues la cantidad total de las que se dan en el alfabeto español es $27! \cong 1{,}09 \times 10^{28} \cong 2^{93}$, pero es muy fácil romper este código porque mantiene las frecuencias.[1]

Entonces, si el método no presenta vulnerabilidades, lo único que queda es probar todas las combinaciones posibles de claves. Pero diversos métodos modernos utilizan claves de 128 bits, así que la cantidad de combinaciones es:

$2^{128} = 3{,}4 \times 10^{38} = 340.282.366.920.938.463.463.374.607.431.768.211.456$

¡Un número mayor que la cantidad de átomos del universo! Obviamente es imposible intentar este camino, aun con las supercomputadoras más rápidas del mundo.

El DES fue quebrado por diversas razones. Una de ellas es que la clave de 56 bits da una cantidad de combinaciones susceptibles de ser atacadas con las computadoras normales. Pero el DES presenta una propiedad curiosa e importante que se conoce como "Triple DES" o "3DES de dos" o "tres claves distintas". Supongamos que elegimos tres claves diferentes, k_1, k_2 y k_3, y encriptamos un mensaje con k_1; luego hacemos como que desencriptamos con k_2, y finalmente encriptamos el resultado con k_3. El que recibe el mensaje hace las mismas operaciones en sentido inverso, o sea que desencripta con k_3, encripta con k_2 y desencripta con k_1, y obtiene el mensaje original.

Si usamos tres claves distintas de 56 bits, la complejidad es 56 x 3 = 168 bits, mucho mayor que los 128 antes mencionados. Pero la pregunta crucial es: ¿existirá otra clave k_4 que dé el mismo resultado que cuando usamos las tres distintas? Si este fuese el caso, todo el procedimiento sería inútil. Sin embargo, se demostró que esa cuarta clave no existe y que, por lo tanto, el 3DES es un método muy seguro (si por ejemplo elegimos $k_1 = k_3$ tenemos dos claves distintas y la complejidad es de 112 bits).

1. El símbolo ! significa "factorial".

Este resultado matemático fue ideal para permitir que el sistema bancario, que había pagado grandes sumas de dinero en dispositivos DES en cajeros, etc., pudiera salvar su inversión cuando el gobierno de los Estados Unidos quiso remover el DES como estándar debido a que había sido quebrado: adaptar ese hardware para implementar el 3DES les permitió ahorrar muchísimo.

Sin embargo, hay debilidades heredadas por el 3DES que pueden ser explotadas por un atacante para reducir su complejidad a unos 80 bits.

¿CUÁLES SON LOS MÉTODOS SIMÉTRICOS MODERNOS?

La evolución tecnológica impuso nuevas demandas en materia de métodos de encriptación. Ya no se pedía únicamente que los mecanismos fueran seguros sino que, además, debían ser muy veloces y adaptables a diversos dispositivos.

Un ejemplo es la televisión codificada: para poder ver una transmisión es necesario decodificar la señal a una velocidad de no menos de 4 Mb por segundo. También la encriptación de grandes bases de datos requiere algoritmos muy rápidos.

Por todos estos motivos, en 2001 el NIST realizó un concurso internacional, del cual resultó ganador el método Rijndael,

acrónimo derivado del nombre de sus autores, los criptógrafos belgas Joan Daemen y Vincent Rijmen, bautizado más fácilmente como AES (*Advanced Encryption Standard*; estándar de encriptación avanzado), que reemplazó el DES y fue especificado en la norma FIPS 197.

Otros algoritmos finalistas del concurso fueron Twofish, MARS, Serpent y RC6.

El AES tiene tres tamaños posibles de claves –128, 192 y 256 bits– y las cantidades correspondientes de claves posibles son $3{,}4 \times 10^{38}$; $6{,}2 \times 10^{57}$, y $1{,}1 \times 10^{77}$.

Debido a su complejidad y al hecho de que ha sido exhaustivamente estudiado por la comunidad criptográfica internacional, es esperable que dure como estándar durante al menos veinte años aunque, como se dice en los estudios de prospectiva, lo único difícil de predecir es el futuro.

Los lectores con formación matemática interesados en la estructura interna del algoritmo AES pueden consultarla por ejemplo en <cs.ucsb.edu/~koc/ns/docs/slides/05-aes.pdf>.

Una condición del concurso internacional en el que resultó elegido el Rijndael fue que se hiciese público y de libre disponibilidad. Por ejemplo, los usuarios de Windows de 64 bits lo pueden descargar gratuitamente de <descargar.cnet.com> y, en el caso de los sistemas de 32 bits, de <descargar.cnet.com>.

Otra opción muy recomendable para usuarios de Linux, Mac, Windows, iPad, iPhone, etc., es <www.aescrypt.com>. Este software se integra al menú contextual de Windows, es decir, cuando uno hace clic con el botón derecho del ratón sobre el nombre de un archivo aparece la opción de encriptarlo con AES. También se integra a la interfaz gráfica de Linux.

¿ESTOS MÉTODOS SON SEGUROS?

Una conclusión fundamental es que actualmente existen métodos al alcance de cualquier persona que generan códigos inviolables, siempre y cuando estén correctamente implementados (es decir, sin puertas traseras [*backdoors*]) y se utilicen claves no triviales (véase la sección que sigue). Además, se pueden combinar métodos: por ejemplo, encriptar un mensaje con el AES 256 y luego volviendo a encriptar el resultado con otro algoritmo como Twofish, Serpent, etc. También se puede hacer algo similar a 3DES con tres métodos distintos.

¿CÓMO EVALUAR UN MÉTODO SIMÉTRICO?

Para evaluar un método simétrico hay que tener en cuenta algunos detalles:

- En primer lugar. si tenemos un texto de, por ejemplo, 1000 caracteres, el resultado al encriptarlo debe tener igual longitud puesto que si es más largo, bien podría ser que el programa esté capturando la clave que usamos y la agregue codificada de alguna manera al texto. Pero también podría ser que el software primero comprima el texto original para "generar espacio" y luego adose nuestra clave. Por eso conviene escribir un texto, comprimirlo con Winzip, Winrar, Tar, tar.gz, Rar, etc., dependiendo del sistema operativo utilizado, ver su longitud y luego encriptarlo. De ese modo no hay "lugar" para agregarle una clave.

- Generar un mensaje que solo contenga una letra repetida (por ejemplo un millón de "a"), encriptarla y luego contar la frecuencia de aparición de cada símbolo (esto obviamente requiere programación).
- Existen otros tipos de test, como el de Maurer, pero ese es un trabajo de especialistas.

EL *HASHING*

La idea es tener un sistema que permita detectar si un mensaje ha sido modificado. Concretamente, una función de *hashing* recibe como entrada una cadena de símbolos y da como resultado otra cadena de longitud fija, de modo tal que, si se altera algún carácter de la cadena de entrada, el nuevo resultado será absolutamente distinto del original. Veamos un ejemplo.

Para el mensaje de entrada "¿ Qué es la seguridad informática?", una función de *hashing* (hay muchas) es SHA-1 [*Secure Hash Algorithm*; algoritmo de hash seguro] y da como resultado "EF8DE7499481DF3C82C05C933CB6CF9DC53E648D".

Supongamos que modificamos el mensaje original eliminando el espacio entre "¿" y "Qué" y obtenemos "¿Qué es la seguridad informática?". El nuevo *hashing* es "44DCFFEC9E70B50DC3C7C6258CDD32B4200C158B", es decir, absolutamente distinto del anterior.

De modo que si, por ejemplo, calculamos un *hashing* de un mensaje, lo encriptamos y mandamos por correo electrónico el mensaje más su *hashing* encriptado, el destinatario podrá, al desencriptarlo, comprobar si se corresponde con lo recibido. En otras palabras, un *hacker* podría interceptar el mensaje y recalcular su *hashing*, pero no le daría igual al que desencriptó el receptor, lo que le impediría cometer un fraude.

Hay que notar que el resultado de un *hashing* tiene una longitud que es independiente del mensaje de entrada. O sea que el de una enciclopedia tiene la misma cantidad de caracteres que el de la letra "a".

Dado que hay muchísimos más mensajes posibles que la cantidad de combinaciones de caracteres del *hashing* final, resulta obvio que existirán muchos mensajes que darán el mismo resultado. A eso se lo denomina "colisión". Al tratar el tema de las firmas digitales veremos que –al menos potencialmente– las colisiones podrían dar lugar a delitos. Por ese motivo se han inventado nuevas funciones de *hashing* que en particular incrementan la cantidad de caracteres del resultado final, algo que dificulta mucho el cálculo de colisiones.

Las aplicaciones son múltiples:

- Al publicar un archivo en la Web (música, libros, software) junto con su *hashing*, un usuario lo descarga, calcula el *hashing* y lo compara con el publicado para ver si fue alterado.
- Informática forense: se calcula el *hashing* del contenido de un disco secuestrado por la Justicia para demostrar luego que al analizar una copia bit a bit de este nada ha sido alterado.
- Información biométrica: por ejemplo, un lector de huellas digitales identifica unos ciento cincuenta puntos clave de un dedo, determina distancias entre ellos y calcula un *hashing* de la imagen para compararlo con una base de datos.

- Generación de números seudoaleatorios a partir de, por ejemplo, los contenidos de memoria de una computadora.
- Los sistemas que escuchan una pieza musical y la identifican lo hacen a través de sus *hashings*.
- Los *hashings* juegan un rol crucial en la firma digital, como veremos más adelante.
- También sirven para la autenticación remota de usuarios, como veremos en la próxima sección.
- Se utilizan en las monedas virtuales, como Bitcoin.

Los *hashings* MD4 (1990) y MD5 (1992) fueron muy usados en los sistemas bancarios, aunque se han dejado progresivamente de lado por su debilidad. En la banca europea se utilizan los RIPEMD (de 128 hasta 320 bits de longitud, del año 1996). El gobierno de los Estados Unidos fijó el SHA-1 como estándar en 1995. Luego, en el año 2002, aparecieron los SHA de 256, 384 y 512 bits. En 2013 apareció el SHA-3.

MANEJO DE CONTRASEÑAS

Normalmente las computadoras utilizan la identificación de los usuarios mediante claves de acceso o contraseñas, y también es común que coexistan diversas cuentas en una misma máquina. Por ejemplo, si se quiere ingresar como administrador, debe haber una clave distinta para él y otras para cada uno de los usuarios.

La identificación de los usuarios mediante *passwords* se conoce en criptografía como "identificación débil". Además, existe toda una variedad de métodos alternativos, como los identificadores mediante el reconocimiento de las huellas

digitales, actualmente muy de moda en *notebooks* y teléfonos inteligentes. Un problema con el uso de las huellas digitales es que alguien puede fabricar dedos de goma con las huellas que una persona haya dejado, por ejemplo, al usar un vaso. Son las llamadas "huellas latentes". También pueden fotocopiarse las huellas desde un documento de identidad y "transportarlas" a láminas transparentes que luego se apoyarán sobre el dispositivo de reconocimiento, o hacer moldes en siliconas, etc. En la Web hay mucho material y videos acerca de este tema (busquen en inglés *fake fingerprints*, o en castellano "copia de huellas digitales"). Eso condujo a diseñar detectores de "dedo vivo" capaces de distinguir entre una simple imagen y un dedo en el cual hay circulación sanguínea. Un caso macabro sucedió en España, donde los jubilados cobran sus haberes mediante el reconocimiento de sus huellas digitales. Los parientes de un anciano jubilado muerto conservaron uno de sus dedos en un *freezer* y lo descongelaban una vez por mes para ir a los cajeros automáticos a cobrar.

Un método mucho más seguro es el reconocimiento del iris, aunque por el momento, debido a su elevado costo, se restringe a instalaciones que requieran mucha seguridad. Asimismo existen otras técnicas, como el reconocimiento de firmas sobre dispositivos que miden características inherentes al firmante; por ejemplo, la presión ejercida y la velocidad (la sola imagen de la firma es fácilmente copiable) o la cadencia de la escritura en un teclado. Esto último proviene del hecho de que una persona adopta patrones de digitación reconocibles al escribir.

Volviendo al tema de la identificación mediante *passwords*, es interesante destacar que aproximadamente el 75% de los delitos informáticos se originan en el uso de claves débiles o fácilmente deducibles por el atacante. Nombres propios vin-

culados al grupo familiar, fechas de cumpleaños, aniversarios, etc., son utilizados frecuentemente por la simple razón de que el usuario los recueda con facilidad.

En la visión del humorista argentino Tute:

Figura 3.1

ES IMPRESIONANTE LA CANTIDAD DE CLAVES QUE TENGO PARA QUE NADIE MÁS QUE YO TENGA ACCESO A UNA CANTIDAD IMPRESIONANTE DE ESTUPIDECES QUE SOLAMENTE A MÍ PUEDEN INTERESARME.

Imagen: cortesía de Tute (www.tutehumor.com.ar).

Y como debemos tener una cantidad creciente de claves para operaciones bancarias, cuentas de correo, redes sociales,

etc., el problema de memorizarlas crece a niveles inmanejables y hace que se generen situaciones de gran vulnerabilidad al utilizar una misma clave para distintas aplicaciones. La solución recomendable es usar un administrador de claves [*password manager*], de los cuales hay una cantidad disponibles en la Web de manera gratuita. Estos programas solo requieren una clave maestra, que es la única que debemos recordar para desencriptar la base de datos que se genera al incorporarle la información de nuestras cuentas con las claves correspondientes. Algo muy útil es que estos programas pueden generar claves al azar de una longitud que fijemos, y que normalmente no usaríamos por ser imposibles de recordar. Por ejemplo, si necesitamos una clave de ocho caracteres para un determinado sitio, un *password manager* puede generarnos algo del tipo "$%/¿23ªv", y ni siquiera debemos escribirla, pues todos tiene una opción para copiarla al *clipboard* (portapapeles), o sea que con un simple clic en el botón derecho del ratón la podemos "pegar" en donde necesitemos hacerlo.

Como tema de seguridad avanzada cabe destacar que es relevante distinguir entre los *password managers* que se comunican a la Web y los que no lo hacen. En el primer caso habría que hacer un estudio técnico para determinar si envían información "adicional" que vulnere nuestro dispositivo o no. En este sentido, es preferible utilizar productos gratuitos que solo generan bases de datos locales. Obviamente estos productos pueden almacenarse en *pen drives* o discos externos para transportarlos adonde queramos.

Une regla mnemotécnica interesante para recordar *passwords* es recitar un poema o una canción y usar la primera letra de cada palabra. Por ejemplo, recordando a Antonio Machado:

Caminante, son tus huellas
el camino y nada más;
caminante, no hay camino,
se hace camino al andar.
Al andar se hace camino,
y al volver la vista atrás...

La clave sería "csthecynmcnhcshcaaaashcyavlva".

Ahora, ¿cuán complejo es quebrar claves probando todas las combinaciones posibles? El cuadro que sigue lo muestra:

Cuadro 3.1

Clave	4 *bytes*	6 *bytes*	8 *bytes*
Minúsculas (26)	0,5 sec	5 min	2,4 días
Minúsculas + dígitos (36)	1,7 sec	36 min	33 días
Caracteres alfanuméricos (62)	15 sec	16 horas	6,9 años
Caracteres ASCII (128)	4,5 min	51 días	2300 años
Bytes (256)	1,2 horas	8,9 años	580.000 años

Exploración a razón de 1.000.000 pruebas/segundo (y el poder computacional se duplica cada 18 meses).

La columna de 4 *bytes* se corresponde con claves de 4 caracteres, por ejemplo, "pepe" si solamente se utilizan minúsculas; si se admiten minúsculas más dígitos podría

ser "pep5", y así. El alfabeto ASCII puede consultarse en <www.asciitable.com> y allí vemos que hay caracteres que normalmente no son usados por las personas por la dificultad para recordarlos, pero que sí pueden generarse con los *password managers*. Así, una clave de ocho *bytes*, o sea de 8 caracteres, que utilice todos los símbolos posibles de los 256 que ofrece el código ASCII extendido, necesitaría 580.000 años de una computadora que efectúe 1.000.000 de pruebas por segundo. Un ejemplo sería "£32aW»Ã≠".

Una pregunta natural es: ¿se pueden realizar 1.000.000 de pruebas por segundo?

Los pasos que deberían seguirse son los siguientes:

- Generar claves probando sistemáticamente todas las combinaciones.
- Intentar desencriptar el archivo codificado con cada clave.
- Conectar el resultado del proceso anterior a un diccionario para ver si aparecen palabras reconocibles.

Aquí hay varias suposiciones: en el primer paso debe definirse la longitud de la clave que se va a probar (hay sitios que siempre definen una longitud de ocho caracteres, algo que les facilita la vida a los *hackers*); en el segundo paso, suponemos que se usó un determinado método de encriptación y, en el tercero, que conocemos el idioma utilizado en el texto que queremos descifrar. Aunque los procesadores actuales pueden ejecutar varios millones de operaciones por segundo, el proceso anterior demanda mucho tiempo, entre otras cosas, por la cantidad de combinaciones que se van a probar y por la longitud del texto a descifrar. Por lo tanto, asumir que uno puede llevar a cabo un millón de pruebas por segundo puede ser

irreal, pero es común utilizar ese parámetro de comparación en criptografía dado que muestra cómo cambia la complejidad del problema según la clase de claves que se utilicen.

ALMACENAR CLAVES DE USUARIOS EN UNA COMPUTADORA

Un escenario normal es que cuando un usuario se conecta a una computadora o una red se identifica mediante un *password* o clave de acceso. Para autenticarlo la computadora no compara la clave introducida con una simple tabla de claves posibles, pues si alguien la *hackea* se acaba la seguridad. Lo que hace –y se utiliza el mismo método desde hace muchísimos años– es almacenar *hashings* de las claves. Entonces, cuando el usuario se conecta e ingresa su clave, el sistema calcula el *hashing* correspondiente y ahí sí lo compara con su tabla interna. De este modo, si alguien obtiene esa tabla, no puede deducir cuáles son las claves que generan *hashings* correctos.

El origen de esta técnica es muy interesante y se describe en el libro *El gato y el ratón* de Clifford Stoll (1991). Stoll fue director del Laboratorio Lawrence en la Universidad de California (Berkeley), y en esa época se les asignaban cuentas a los científicos para cobrarles –al menos virtualmente– por el uso de los recursos computacionales. Una pequeña discrepancia contable de 0,75 dólares lo llevó a investigar lo que sucedía, y a darse cuenta de que un *hacker* entraba al sistema. En vez de bloquearlo, lo dejó actuar para ver lo que hacía, e incluso llegó a conectar a la computadora una "impresora esclava" cuya única función era imprimir en papel todos los comandos que utilizaba el intruso.

En esa época los sistemas Unix tenían dos versiones: la desarrollada en Berkeley y el llamado System V, originado en la Costa Este de los Estados Unidos. El intruso usaba System V, o sea que era alguien "foráneo".

Stoll acudió al FBI y la CIA, entre otros organismos, sin lograr ningún resultado. Luego de una ardua y fascinante investigación resultó que se trataba de una red de espionaje de Alemania Oriental (instalada en Hannover) que utilizaba centrales telefónicas electromecánicas que no dejaban rastro para introducirse a la Mitre Corporation, una importante empresa de la Costa Este dedicada a asuntos de Defensa. De allí "saltaban" a Berkeley, pero no para escudriñar a los científicos, sino para construir un diccionario de *hashings*. Es decir que introducían contraseñas para ver en qué derivaban en el sistema de computación. Más aún, pusieron todas las palabras de un diccionario inglés para construir una tabla completa.

El siguiente paso fue pasar del laboratorio Lawrence a sistemas de Defensa para capturar las tablas de *hashings* de claves. Un ejemplo imaginario: la palabra "*river*" tiene como *hashing* MD4 "795DBD018B7A539ECD50435CA6BA9C6E". Entonces, si encontraban ese *hashing* en una tabla, sabían que la clave para entrar era *river*.

De allí surgió la norma de usar letras y números en forma obligatoria pues *river* se encuentra en un diccionario inglés, pero "river28" no, con lo cual se impide el llamado "ataque del diccionario".

AUTENTICACIÓN REMOTA

La autenticación remota es un tema crítico en el mundo actual. Las personas deben conectarse a su computadora

personal, a dispositivos diversos, al sistema de sus lugares de trabajo, al *home banking*, a las redes sociales, etc., y vimos que la identificación mediante *passwords* se llama "débil".

El paso siguiente en la seguridad lo constituye la autenticación de dos factores: algo que sabemos y algo que poseemos. Por ejemplo, los *tokens* electrónicos son pequeños dispositivos (lo que poseemos) que pueden activarse mediante una clave (lo que ya sabemos), generando un número que se puede "pegar" en un formulario web.[2] El servidor, o sea la máquina de la cual depende ese formulario, puede calcular dicho número de acuerdo con la identificación reclamada por el usuario. Si ambos coinciden, el usuario queda autenticado. La ventaja es que los números generados varían cada vez, por lo que pueden considerarse *passwords* de "única vez" o *passwords* descartables que un *hacker* no puede volver a utilizar aunque los intercepte.

Originalmente los números se generaban sincronizando los relojes del servidor y del *token* debido a que el mecanismo de cálculo dependía de la hora de un momento dado. Luego se vio que esos relojes diferían fácilmente pues el del *token* era en general algo muy elemental y poco confiable. Por eso, una solución fue conectarse a través de la Web con relojes atómicos, algo muy simple que solo requiere un *ping*.[3]

2. Véase, por ejemplo, "*Token* de seguridad", en <www.es.wikipedia.org>.

3. *Ping* es un procedimiento que consiste en enviar un paquete de información a una dirección IP y medir el tiempo que tarda en ir y volver. Una referencia es el programa Neutron, disponible en <keir.net>, para usuarios Windows. Para los que usan Linux véase, por ejemplo, el video "Sincronización Ubuntu Linux con reloj atómico mediante NTP", en YouTube. Una referencia interesante —entre muchas— es "¿Qué es un *reloj atómico*?", en <www.ordenadores-y-portatiles.com>.

Un esquema más simple, y que es el que predomina hoy en día, es la sincronización por eventos. La idea es que si el usuario por ejemplo generó 23 números, el servidor sabe cuál será el 24. Entonces, una persona recibe un *token*, que es único en su configuración, y, cuando se conecta a la Web para hacer una transacción, el sistema le pide que genere un número (hay modelos muy prácticos en los cuales solamente es necesario abrirlos, ver el número que aparece en una pantalla, y cerrarlos, con lo que quedan listos para la siguiente operación). Esos números tienen una validez acotada en el tiempo, por ejemplo, un minuto, y el servidor no acepta *replay*, o sea, volverlos a utilizar aun antes de su vencimiento. Si el usuario quiere realizar más de una operación en menos de un minuto, debe generar nuevos números. La experiencia también ha mostrado que si un niño juega a abrir y cerrar el *token* de alguno de sus progenitores, el dispositivo perdía la sincronización con el servidor. Por ello, actualmente, si un servidor recibe un número que no es el siguiente en la secuencia normal, procede a calcular por ejemplo los cien números posteriores para ver si alguno coincide. Si ese es el caso, resincroniza todo y vuelve a la normalidad y, si no, rechaza la transacción asumiendo que alguien intenta cometer un fraude.

Asimismo existen muchos otros métodos en la criptografía para autenticar, e incluso hay libros enteros dedicados al tema, como *Authentication: From Passwords to Public Keys*, de Richard Smith (2002).

Veamos algunos de ellos:

- El verificador, o sea, la computadora o sistema dedicado a identificar al usuario, le manda un mensaje pidién-

dole que lo encripte con su clave secreta. El usuario lo hace y devuelve el mensaje encriptado. El verificador ve si se corresponde con el uso de la clave correcta. Esa es la técnica de los MAC [*Message Authentication Codes*; códigos de autenticación de mensajes].

- Los transponedores en la aviación son una mezcla de receptores y emisores que cumplen el propósito de determinar sus posiciones para evitar colisiones, pero también para identificarse. Modernamente la ADS-B es una señal en UHF emitida por un avión que incluye a la vez su identificación y su posición.
- Los métodos biométricos (huellas digitales, reconocimiento del iris o de rostros).
- Los métodos de clave pública que veremos en el próximo capítulo.
- Los protocolos basados en *hashings*, como el de Lamport, que describiremos a continuación.

El método de Lamport

Un usuario recibe inicialmente una clave secreta y, cuando quiere autenticarse, el sistema le pide que por ejemplo calcule una cantidad de *hashings* sucesivos de la clave. Supongamos que la clave es "Don Quijote 53".

Su *hashing* SHA-1 es "F642D121390F08465513A C0A95D7123F" y el *hashing* de este *hashing* resulta ser "515D1A23F9970F1E1AA4E3DA4152E8FE".

El verificador recibe el último *hashing* y comprueba que su origen era la clave secreta. Un *hacker* que lo intercepte no podrá calcular de dónde provino.

¿CUÁL ES LA GRAN DEBILIDAD DE LOS MÉTODOS SIMÉTRICOS?

Recordemos que los métodos simétricos se caracterizan por usar la misma clave para encriptar y desencriptar, o sea que para que dos entidades (personas u organizaciones) puedan comunicarse es imprescindible que la compartan.

Supongamos que el emisor y el destinatario se encuentran personalmente, acuerdan una clave y luego se separan. Si en algún momento deben cambiar la clave, por razones de seguridad o porque esta se vio comprometida, y están separados geográficamente y sus medios de comunicación han sido penetrados por espías, aparece la cuestión de cómo acordar una nueva clave sin que sea interceptada.

Este ha sido el eterno problema de la transmisión de información sensible, que no solo genera vulnerabilidades, como en el caso de las Enigma, sino que acarrea grandes costos, como mandar emisarios a las embajadas con frecuencia, etc., porque obviamente los mensajeros pueden ser interceptados también.

¿Qué es la criptografía de clave pública?

Acerca de rompecabezas, platos rotos, números primos y una pizca de azar

La criptografía de clave pública se considera un enorme logro científico. Su utilidad es ubicua:

- Resuelve el problema de la distribución segura de claves.
- Todo el comercio electrónico a través de la Web se basa en ella, y en 2012 el monto mundial llegó al trillón de dólares (eMarketer, 2013).
- Permite "firmar" documentos en forma digital, por ejemplo, contratos amparados en las leyes de firma digital de numerosos países.
- Con las cadenas de confianza adecuadas (usando certificados digitales emitidos por autoridades certificantes) posibilita todo tipo de operaciones entre entidades de diversos países.
- Permite autenticar los sitios de la Web, convirtiéndolos en sitios seguros para realizar compras, transacciones bancarias, etc.

Veamos algunos detalles y la historia de estos métodos.

Supongamos que dos entidades se comunican por algún medio, sea correo electrónico, teléfono, etc., que se encuentra totalmente intervenido por espías que registran todo lo que dicen o escriben. ¿Pueden acordar una nueva clave sin que sea detectada por los espías? Aunque el problema aparenta ser insoluble, curiosamente es factible hacerlo.

En la criptografía de clave pública cada entidad posee una clave privada y una pública que funcionan como "inversas" entre sí. Esta última, como su nombre lo indica, puede ser conocida por todo el mundo. Cuando una entidad encripta un mensaje con su clave privada, puede ser descifrado mediante su clave pública. En ese caso no hay ningún secreto pues cualquiera puede tener acceso a esa clave pública. Pero al descifrarla en forma coherente queda claro que el mensaje original había sido encriptado con la clave privada del emisor, o sea que se lo ha autenticado. Esta es la base de la firma digital.

Ahora, si encriptamos un mensaje con la clave pública del destinatario, la única entidad que lo puede descifrar es el dueño de la correspondiente clave privada, o sea, el receptor. Con este mecanismo simple le podemos enviar un mensaje con una nueva clave de un método simétrico, como el AES, y nadie podrá interceptarlo (valen las mismas observaciones acerca de que los programas que utilicemos no tengan puertas traseras, etc., tema que desarrollaremos luego con más detalles).

La criptografía de clave pública y la asimétrica son distintas, aunque están muy relacionadas. En la primera la clave pública se puede difundir sin problemas, pero en la segunda se conserva un secreto adicional.

Durante casi tres décadas se consideró que Whitfield Diffie y Martin Hellman habían sido los iniciadores de la

criptografía asimétrica, pero distintos libros le atribuyeron a un joven estudiante de Berkeley, Ralph C. Merkle, el haber descubierto en forma independiente la criptografía de clave pública a través de los rompecabezas que llevan su nombre. La motivación de todos ellos era la misma: en la clásica criptografía simétrica el emisor y el receptor comparten una misma clave que debe cambiarse periódicamente por razones de seguridad. El problema era entonces cómo comunicar a través de canales inseguros la nueva clave.

Sin embargo, la historia volvió a dar un giro cuando el gobierno británico decidió hacer público que el sistema había sido desarrollado en el Cuartel General de Comunicaciones del Gobierno [Government Communications Headquarters; GCHQ] unos cuantos años antes. ¿Por qué motivo no lo habían difundido en su momento? Por las mismas razones por las cuales destruyeron las máquinas Colossus y toda la documentación de Bletchley Park al finalizar la guerra: el secreto militar.

Con la proliferación de las tecnologías de las comunicaciones la cuestión de distribuir continuamente claves a las fuerzas armadas, cuerpo diplomático, etc., se había transformado en una pesadilla logística que presentaba serios problemas de seguridad.

La primera idea la tuvo James H. Ellis, quien trabajaba en el GCHQ y a quien le habían encargado estudiar el tema de la distribución de claves en forma segura a principios de los años sesenta. Según él, la solución se le ocurrió al tener acceso a un documento de autor desconocido acerca de un teléfono militar de la empresa Bell, donde se describía un método ingenioso para volver irreconocibles los mensajes hablados. Consistía en que el receptor agregara ruido a la línea de comunicación, que luego podría eliminar para recu-

perar el mensaje original. El método no era práctico, pero incorporaba una idea nueva: el receptor del mensaje debía intervenir en el proceso de encriptación agregando el ruido que solo él conocía. Un espía únicamente escucharía ruidos sin sentido.

Las ideas de Ellis eran análogas a las que tuvieron Diffie, Hellman y Merkle varios años después, pero ni él ni sus jefes sabían cómo implementarlas en la realidad. Sin embargo, en 1973 un joven matemático de la universidad de Cambridge, Clifford Cocks, se unió al grupo. No sabía nada de criptografía, pero sí de una disciplina matemática que condujo a la solución del problema: la teoría de números.

Cocks comenzó a pensar en las llamadas "funciones de una única vía", en las cuales una operación en un sentido es fácil de hacer, pero su inversa no. Un ejemplo no matemático sería romper un plato en pedazos minúsculos, algo simple de lograr, pero su "inversa", o sea reconstruir el plato, es algo muy complejo. En el terreno matemático un candidato natural para esa tarea es la multiplicación de números naturales y el problema de factorizar el resultado. Por ejemplo, si consideramos el número 89.031.567.113, no es obvio que se puede escribir como 13.649 x 6.522.937, pero es fácil multiplicarlos.

Si el número es mucho más complejo, en general nos enfrentamos con el problema de la factorización de enteros, el cual, hasta el momento, no ha sido resuelto. De hecho existen desafíos que nadie pudo resolver, un tema que trataremos nuevamente al discutir la introducción de puertas traseras como método de espionaje.

Cocks recuerda que encontrar la solución al problema le llevó nada más que media hora. Lo que hizo era lo que años

más tarde se conocería como el "método RSA" (por las iniciales de los apellidos de sus creadores: Ronald Rivest, Adi Shamir y Leonard Adleman). Cocks no se daba cuenta de las tremendas consecuencias de su descubrimiento y, como tenía implicaciones revolucionarias desde el punto de vista militar, fue mantenido en el máximo secreto.

En 1974 otro criptógrafo, Malcolm John Williamson, se unió al equipo. Cocks le explicó su solución y, como a Williamson no lo convencía, se abocó a tratar de encontrarle defectos. No lo logró, pero en cambio desarrolló lo que hoy en día se conoce como el "método de Diffie-Hellman".

¿Qué pasó del "lado estadounidense"? Diffie, quien estaba obsesionado con la distribución segura de claves, daba conferencias sobre el tema sin convencer a sus oyentes dado que no presentaba un algoritmo concreto. Mientras dictaba una en el laboratorio Thomas Watson de IBM conoció a Alan Konheim, un criptógrafo respetado que le mencionó que un profesor de Stanford, Martin Hellman, había hablado recientemente allí sobre esa cuestión.

Diffie emprendió inmediatamente el largo viaje hacia Palo Alto para conocer a Hellman y así comenzaron una relación académica muy fructífera. Pero seguían sin encontrar una solución.

A diferencia de Cocks, ninguno de los dos tenía conocimientos de teoría de números, según me dijo Hellman en una reunión que tuvimos el 20 de julio de 2001. De hecho, un profesor de Stanford, cuando escuchó lo que pretendían resolver les había aconsejado que pensaran en números primos, pero ninguno de los dos entendía por qué.

En 2007, Hellman me comentó que todavía estaba sorprendido de lo que habían logrado con Diffie en esa época,

considerando que entonces sabían muy poco de teoría de números. Esto vuelve su trabajo todavía más meritorio.

EL MÉTODO DE DIFFIE-HELLMAN DE INTERCAMBIO DE CLAVES

Para entenderlo necesitamos recordar los conceptos de la aritmética modular que vimos anteriormente, al tratar el método de Julio César.

El objetivo del método de Diffie-Hellman es que dos interlocutores (usuario 1 y usuario 2) lleguen a un número en común que no pueda ser detectado por un espía, al que deberán darle un significado. Una posibilidad es usar ese número como clave, que designe una palabra en un libro, etc. Para ello se elige un primo muy grande *p* (no menos de 150 dígitos decimales) y un entero *a* menor que *p* (idealmente *a* debe tener una propiedad adicional, pero que dejaremos de lado por ser un tema muy matemático y porque el método igual funciona aunque esta no se cumpla).

Vamos a ponernos en el peor de los escenarios, o sea, que el espía sabe cuáles son los valores de *a* y *p*.

Ahora viene lo ingenioso: el usuario 1 se imagina un entero *x*, y el usuario 2 otro entero *y* que nunca se pondrán en evidencia.

El usuario 1 calcula $z_1 = a^x \pmod{p}$ y se lo manda al usuario 2.

El usuario 2 calcula $z_2 = a^y \pmod{p}$ y se lo manda al usuario 1.

Finalmente el usuario 1 calcula $K = z_2^x \pmod{p}$ y el usuario 2 calcula $z_1^y \pmod{p}$, ¡que resulta ser también *K*!

Este es el simple y extraordinario resultado que permite que ambos usuarios arriben al número en común *K*.

El espía no conoce ni *x* ni *y* así que le es imposible obtener *K* si *p* es muy grande. Veamos un simple ejemplo numérico. Sea p = 23, a = 5, x = 6 (imaginado por el usuario 1) e y = 10 (imaginado por el usuario 2):

$z_1 = a^x \pmod{p} = 5^6 \pmod{23} = 15625 \pmod{23} = 8$
y se lo envía al usuario 2
$z_2 = a^y \pmod{p} = 5^{10} \pmod{23} = 9765625 \pmod{23} = 9$
y se lo envía al usuario 1
$z_2^x \pmod{p} = 9^6 \pmod{23} = 3$

$z_1^y \pmod{p} = 8^{10} \pmod{23} = 3$

Por lo tanto, ¡ambos usuarios llegaron a la misma clave K = 3!

A esa clave le darán un significado; por ejemplo, usar como clave de encriptación el tercer párrafo de la página 7 de un periódico que se publicará al día siguiente.

El espía, que interceptó la comunicación, determinó que el usuario 1 le mandó al usuario 2 el número 8. Como conocía *a* y *p*, para calcular *x* solo tiene que resolver $5^x = 8 \pmod{23}$, lo cual es trivial para números pequeños. Veamos qué sucedería con unos números apenas más grandes:

p = 1.000.475.149
a = 876.543.098
x = 12.345.678.909.876.543

En ese caso la ecuación final para encontrar *x* con la que debería lidiar el espía es:

$876543098^x \pmod{1000475149} = 993617947$

Esto da una idea de cómo crece la complejidad cuando se toman números más grandes. Este es el problema del logaritmo discreto, el cual –y a pesar de un avance teórico reciente en 2014– todavía carece de un método eficaz para ser resuelto.

Los equipos de comunicaciones como los *routers* continuamente intercambian claves en forma autónoma, o sea, sin intervención humana, para encriptar sus comunicaciones. En la práctica esto significa que pueden usar claves continuamente cambiantes, lo cual es un adelanto de seguridad gigantesco con respecto a los métodos tradicionales.

EL MÉTODO RSA

Este método es el más famoso de los de clave pública y fue, y todavía es, un estándar de facto que se transformó en el algoritmo criptográfico más relevante de la historia. Veamos en qué consiste:

1. Elegir dos números primos "grandes" y distintos, sean *p* y *q* (luego mencionaremos cómo obtenerlos).
2. Calcular $n = p \times q$.
3. Obtener un entero *c* coprimo con $(p-1) \times (q-1)$, o sea que el máximo común divisor sea 1. Por ejemplo, se puede elegir *c* primo siempre que no sea un divisor de $(p-1) \times (q-1)$.
4. Calcular el único entero *d* que cumple la ecuación $e \times d = 1 \bmod ((p-1) \times (q-1))$.
5. Destruir *p, q,* $(p-1) \times (q-1)$.
6. La clave pública es *(d, n)* y la privada *(c, n)*, o viceversa.

Para encriptar un mensaje *M* (entendido como un número entero) se calcula $M_1 = M^c \pmod n$; y para desencriptarlo $M = M_1^d \pmod n$. M_1 es la firma digital del mensaje *M* realizada con la clave privada *(c, n)* del usuario.

Por ejemplo: sea p = 127, q = 103, n = 127 x 103 = 13081, (p-1) x (q-1) = 12852.

Elegimos c = 59 (primo) y la ecuación del punto 4 anterior resulta 59 x d = 1 (mod 12852), cuya solución es d = 1307.

Supongamos que queremos transmitir la letra "b", segunda letra del alfabeto. Nuestro mensaje es: $M = 2$. Entonces, $M_1 = M^c \pmod n = 2^{59} \pmod{13081} = 10041$, que es lo que se transmite. El receptor posee la clave pública del emisor, que es *(1307, 13081)*, así que puede calcular $M_1^d \pmod n = 10041^{1307} \pmod{13081} = 2$, recuperando el mensaje original.

Decimos entonces que 10041 es la firma digital de M = 2 realizada con la clave privada del emisor.

Obviamente como la clave pública contiene al módulo *n*, si es posible factorizarlo, entonces se obtienen los primos *p, q* y a partir de ello, y usando el entero *d*, se puede resolver la ecuación del punto 4 obteniendo *c*, la clave privada, y quebrando así el sistema. Por eso se utilizan primos muy grandes del mismo orden de magnitud.

Cuando se habla de, por ejemplo, RSA1024 quiere decir que el módulo *n* tiene una longitud de 1024 bits, lo que equivale a 310 dígitos decimales, o sea que los primos *p, q* tienen 155 dígitos cada uno.

Debido al éxito de esta tecnología de clave pública se creó la empresa RSA Security, adquirida luego en 2006 por la corporación EMC por 2,1 billones de dólares. En 1991 los

laboratorios de RSA Security establecieron los desafíos de la factorización, una suerte de concurso con problemas de distinta longitud en el que se entregaban premios, que fueron retirados en el año 2007.[1]

Hasta ahora el mayor problema que ha sido resuelto es el del RSA768 de 232 dígitos decimales, récord logrado el 12 de diciembre de 2009. Ese número es:

123018668453011775513049495838496272077285356959533479219732245215172640050726365751874520219978646938995647494277406384592519255732630345373154826850791702612214291346167042921431160222124047927473779408066535141959745985690214341 3

Y sus factores son:

p = 33478071698956898786044169848212690817704794983713768568912431388982883793878002287614711652531743087737814467999489

q = 36746043666799590428244633799627952632279158164343087642676032283815739666511279233373417143396810270092798736308917

¿Fue fácil? No, en absoluto, pues el trabajo comenzó en agosto de 2007, e involucró a científicos y equipos de computación de Alemania, Australia, Francia, Inglaterra, Japón y Suiza. Resulta, entonces, que si usamos una clave RSA768 y la cambiamos cada, digamos, tres meses, por el momento

1. Véase, por ejemplo, "RSA Factoring Challenge", en <www.en.wikipedia.org>.

estamos seguros. Sin embargo, lo más común es usar claves RSA2048 que tienen alrededor de 617 dígitos decimales.

El desafío actual más importante, que no ha sido resuelto todavía, es:

RSA1024 = 135066410865995223349603216278805969938881475605667027524485143851526510604859533833940287150571909441798207282164471551373680419703964191743046496589274256239341020864383202110372958725762358509643110564073501508187510676594629205563685529475213500852879416377328533906109750544334999811150056977236890927563

Para implementar este método naturalmente se usa un software especial. Es muy importante su eficacia y velocidad, dado que en muchas aplicaciones la decodificación de las firmas digitales debe hacerse en tiempo real por el gran volumen de transacciones involucradas. Un ejemplo típico son los mercados de valores que reciben transacciones firmadas digitalmente por los agentes financieros, una actividad que suele concentrarse en pocas horas por día. En esos casos es deseable fijar la clave pública en, por ejemplo, el valor 3, o sea, usar pares de la forma $(3, n)$ que no comprometen la seguridad y permiten acelerar el proceso.

OTROS MÉTODOS DE CLAVE PÚBLICA

Existen varios métodos, como el de El Gamal, que posee la propiedad interesante de que distintas encriptaciones de un mismo mensaje varían porque involucran un núme-

ro al azar que es cambiante. Hay otros algoritmos, como el de Rabin, el E-sign y el DSA [*Digital Signature Algorithm*; algoritmo de firma digital] (de este último hablaremos en el apartado "Política y criptografía" del capítulo que sigue).

En la práctica, el método más usado, aparte del RSA, es el de curvas elípticas. El tema es muy arduo matemáticamente hablando, pero basta con mencionar que se logra la misma seguridad de RSA pero con claves mucho más cortas, lo que lo vuelve muy útil para los dispositivos móviles, como los teléfonos inteligentes, ya que requiere menos memoria.

FIRMA DIGITAL

Vimos el concepto de firma digital cuando tratamos el tema del método RSA, pero obviamente es imprescindible asociar una identidad a una clave pública dada pues es la única manera de otorgarle seguridad al sistema.

Hay certificados digitales de diversas clases. La clase 1 sirve solo para identificar una dirección de correo, pero no la identidad de su dueño. Po ejemplo, se pueden obtener certificados gratuitos de clase 1 con el nombre Napoleón Bonaparte. Ahora, si dos personas poseen certificados digitales de clase 1, pueden intercambiar mensajes encriptados y firmados (véase más adelante, el capítulo 5).

Las leyes de firma digital vigentes en numerosos países definen, por un lado, los entes licenciantes, que son los organismos autorizados para otorgar licencias a las autoridades certificantes. Ellas tiene como misión fundamental constatar la identidad de las entidades y asociarlas con una determinada clave pública. En el caso de las personas, consiste en

verificar sus documentos, etc., y, en el de las empresas, sus balances, la conformación de los directorios, etc.

Los certificados de clase 2 son para personas y se emiten luego de que la autoridad certificante haya verificado la identidad para asociarla responsablemente a su clave pública.

Los certificados de clase 3 son para empresas.

La clave pública se "encapsula", junto con los datos de la persona o entidad, en un archivo digital denominado "certificado", que a su vez es firmado digitalmente por la autoridad certificante, la cual debe poseer seguros muy grandes para poder responder ante posibles fraudes.

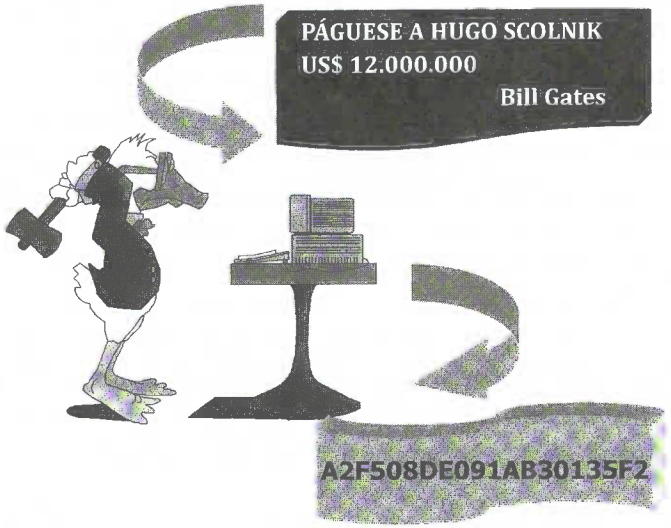
Los certificados digitales se incorporan a los navegadores de modo tal que si una entidad desea comunicarse con otra que también posee un certificado, no solo puede firmar digitalmente su mensaje sino que, además, puede encriptarlo. Esto se realiza mediante el protocolo SSL [*Secure Sockets Layer*; capa de conexión segura], que en esencia hace lo siguiente:

- "Negociar" entre ambos usuarios cuáles son los métodos de encriptación más fuertes que poseen en común.
- Intercambiar sus certificados digitales para tener las respectivas claves públicas.
- Dado que los métodos simétricos, como el AES, son casi cien veces más rápidos que el RSA, parte del intercambio consiste en acordar una clave común para poder encriptar los contenidos del mensaje en forma veloz. Se deja la parte asimétrica, como el RSA, para intercambiar la clave simétrica, firmar los *hashings*, etc.

Veamos un ejemplo completo de cómo funciona la firma digital. Supongamos que un banco recibe un mensaje acom-

pañado por un *hashing*, supuestamente de Bill Gates, que dice que me paguen 12.000.000 de dólares.

Figura 4.1



Naturalmente, el banco no tiene por qué confiar en lo que recibió pues un atacante podría haber interceptado un mensaje real que decía que me paguen 1 dólar, haber procedido a modificarlo, recalcular su *hashing* y enviar todo.

Pero ¿qué sucede si Bill Gates procede a firmar digitalmente el *hashing* de su mensaje original y el banco accede a su certificado de clave pública? Entonces, nos encontramos con la siguiente situación.

Figura 4.2

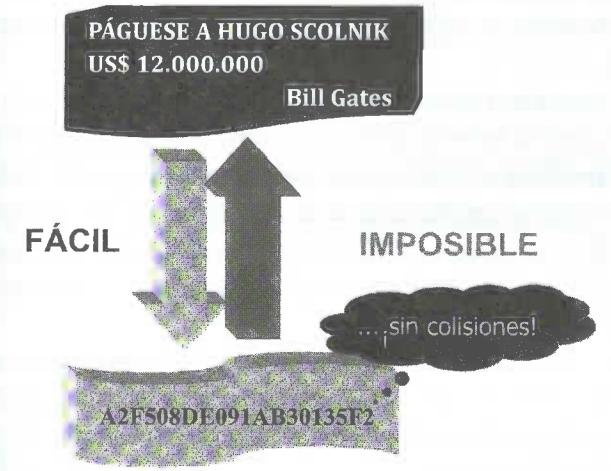
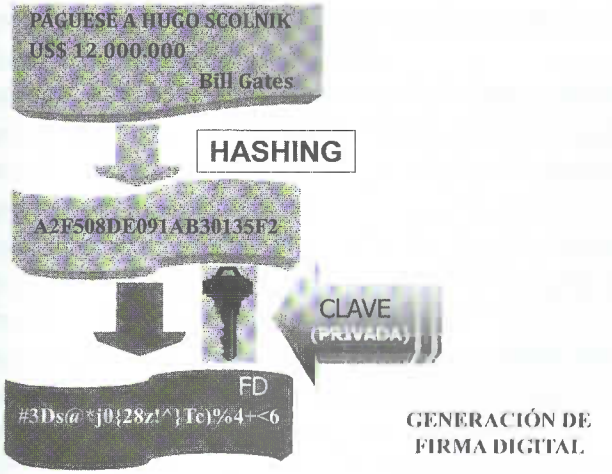


Figura 4.3



En ese caso el mensaje original queda validado, pues el *hacker* nunca habría podido firmar digitalmente ninguna modificación, a menos que posea la clave privada de Bill Gates.

El mensaje fue transmitido en claro (sin encriptar), cualquier podría haberlo leído, pero era imposible modificarlo. En la práctica el protocolo es más complejo, pues el emisor y el receptor acuerdan una clave simétrica, sea por el método de Diffie-Hellman o mediante el uso de las respectivas claves públicas, y con ella encriptan el mensaje usando el AES, 3DES, etc.

Mediante el software gratuito <www.fd.com.ar/download/crypeasy500free.zip> se pueden encriptar archivos, calcular *hashings*, generar claves RSA y hacer operaciones de firma digital sin certificados.

¿Existen los sitios seguros?

Acerca de delincuentes del mundo virtual, candados, puertas traseras y técnicas de encriptación que se consideran armas

Normalmente enviamos los datos por la Web mediante un protocolo llamado HTTP [*Hypertext transfer protocol*; protocolo de transferencia de hipertexto], que no maneja encriptación y, por lo tanto, es vulnerable. En cambio, el SSL conduce al HTTPS [*Hypertext Transfer Protocol Secure*] que sí encripta los datos y es la base del comercio electrónico.

Cuando se busca una página en Internet, esta puede comenzar con "http" (insegura) o "https" (segura). Los sitios seguros son aquellos que poseen un certificado digital y se distinguen por poseer un candado amarillo (sobre el cual hay que cliquear para ver el certificado digital correspondiente, quién lo emitió, cuándo vence, etc.) porque, justamente, su dirección comienza con esa sigla.

Es importante revisar el candado pues en Italia hubo un caso de un *hacker* que puso nada más que la imagen de un candado para simular un sitio seguro.

Para tener una idea de la diferencia entre ambos protocolos, los usuarios de Windows pueden realizar lo siguiente:

- ir a Inicio;
- cliquear Ejecutar;
- escribir "cmd";
- dar Enter;
- escribir "tracert www.ssl.com", y
- apretar Enter.

Este comando va a enlistar todos los sitios que recorrerá, comenzando con su computadora (escribirá su IP). Lo normal es que puedan aparecer hasta treinta sitios recorridos portando su información, y cada uno podrá interceptar sus claves. En cambio, HTTPS las encripta y, por eso, estarán seguras.

¿QUÉ SIGNIFICA CUANDO VEMOS UNA BARRA VERDE EN UN SITIO DE INTERNET?

La barra verde para la dirección en la parte superior de la pantalla indica que el sitio que usted está usando es legítimo y seguro. Es decir, no es un sitio falso donde lo pueden engañar para que revele información privada (véase *phishing* más adelante).

Esto se obtiene con los certificados SSL de servidores web con validación extendida que aseguran una encriptación de al menos 128 bits y con garantías monetarias muy importantes. Son los sitios recomendables.

POLÍTICA Y CRIPTOGRAFÍA

Como mencionamos antes al tratar la introducción del DES, el primer método criptográfico para uso civil, el problema que se suscitaba era el posible fin del monopolio de las organizaciones gubernamentales sobre la encriptación de datos.

La preocupación se centraba primordialmente en la posibilidad de que grupos terroristas o delincuentes usaran técnicas de encriptación. Para ello se propuso desde un principio introducir mecanismos de recuperación de la información mediante las llamadas "puertas traseras". Ante las críticas de los grupos defensores de los derechos civiles, la respuesta era que las personas no involucradas en actividades fuera de la ley no tenían por qué preocuparse de que se conociera esa información.

Antes de continuar, vale la pena destacar que existe mucha información acerca del uso de criptografía por parte de grupos terroristas o delincuentes. Algunos ejemplos:

- El 20 de marzo de 1995 el grupo terrorista Aum Shinrikyo efectuó un ataque con gas sarín en el subterráneo de Tokio. Parte de sus documentos estaban encriptados con RSA.
- Interpol solicitó en su momento la ayuda de diversos criptógrafos para intentar descifrar documentación secuestrada al Cártel de Medellín, liderada por Pablo Emilio Escobar Gaviria. En ellos había referencias a cuentas en Suiza.
- Ramzi Ahmed Yousef, uno de los responsables del ataque a las Torres Gemelas, poseía información encriptada en su computadora.

- Una práctica usual del crimen organizado es tener sus registros contables cifrados.

En los Estados Unidos, hasta 1996, la exportación de criptografía con claves mayores de 40 bits estaba muy limitada. Recientemente, en 2004, el ex director del FBI Louis Freeh, en su testificación en la Comisión del 11S, pidió nuevas leyes contra el uso público de la criptografía. La idea de limitar la longitud de las claves a 40 bits se debía obviamente al hecho de que, en ese caso, es factible quebrarlas por "fuerza bruta".

Hay dos situaciones para destacar en lo referente a las políticas oficiales de los Estados Unidos, que asimilaban la exportación de productos criptográficos al tratado de armas estratégicas,[1] y sobre el acuerdo de Wassenaar,[2] que reunió a muchos países en un intento por recortar el uso de la criptografía. La primera es que cualquier entidad interesada en conseguir encriptación con una fortaleza de claves superior a los 40 bits podía comprarla en diversas fuentes (Alemania, Argentina, Australia, Israel, etc.).

Hasta 1992 la exportación de productos criptográficos con longitud de claves mayores a 40 bits estaba asimilada al Reglamento del Tráfico Internacional de Armas [*International Traffic in Arms Regulations*; ITAR], o sea que se consideraba equivalente a la venta de armamentos. Aunque desde ese año disminuyeron las restricciones, varias aún persisten.

El segundo aspecto que debe tomarse en cuenta es que debido a que los algoritmos –al menos los usuales– son

1. Véase, por ejemplo, "Criptografía: mismo reglamento que un arma en tratados internacionales", en <www.kriptopolis.com>.

2. Véase <www.wassenaar.org>.

públicos, muchas personas pueden programarlos. Por ejemplo, el método RSA requiere bibliotecas especiales para manejar números de gran tamaño que están disponibles en diversas fuentes, como <www.gmplib.org>. Programar RSA es un ejercicio habitual en nuestros cursos de criptografía y, aunque el código resultante no sea competitivo en velocidad con los productos comerciales, le permite al usuario encriptar mensajes con un altísimo grado de seguridad, por ejemplo, con claves de 4096 bits.

Un caso muy actual se encuentra en Recorded Future (2014), donde vemos que la organización responsable del ataque a las Torres Gemelas ha desarrollado sus propios productos de encriptación.

Las controversias acerca del conflicto entre la deseable privacidad de los individuos y las políticas restrictivas estadounidenses tuvieron un protagonista notable: Phil Zimmermann, considerado el adalid de la criptografía fuerte para uso público. Fue quien implementó el método RSA, el que publicó en 1991 el código fuente de lo que bautizó como PGP [*Pretty Good Privacy*; privacidad bastante buena] y quien dijo: "Si la intimidad está al margen de la ley, solo los que estén al margen de la ley tendrán intimidad".

Ante la amenaza de una legislación que estaba por ser establecida por el gobierno de los Estados Unidos e iba a exigir la implantación de puertas traseras en todos los productos criptográficos desarrollados dentro del país, Zimmermann distribuyó una versión libre de PGP. Sus esfuerzos para publicar PGP en todo el mundo le granjearon una larga batalla con el Departamento de Justicia por la supuesta violación de las restricciones de exportación, hecho originado a partir de una controversia con la empresa RSA Data Security Inc. por

el uso del algoritmo. Finalmente, el Departamento de Justicia abandonó el caso contra Zimmermann, la distribución libre de PGP fue factible a nivel internacional y terminó convirtiéndose en un estándar abierto [*OpenPGP*].

Un problema con PGP es el manejo de las claves públicas y su asociación con la verdadera identidad de una persona. Las claves pueden buscarse en <keyserver.pgp.com/vkd/GetWelcomeScreen.event> y todo el sistema tiene utilidad en grupos cerrados donde existe un nivel de conocimiento y confianza personales.

Pero no es lo mismo si una persona en un país A quiere hacer una transacción con otra de un país B sin que existan elementos convincentes acera de las respectivas identidades. Justamente los certificados digitales emitidos por autoridades certificantes autorizadas por las legislaciones de sus correspondientes países es lo que permite validar operaciones en forma confiable.

También había otro actor muy importante en el escenario internacional: el comercio electrónico [*e-commerce*]. La razón es que las compañías interesadas en vender sus productos a través de la Web necesitaban con urgencia disponer de métodos seguros para firmar digitalmente las transacciones, pero eso estaba en contradicción con las limitaciones derivadas de la seguridad nacional. Esto generó una gran presión para que se desarrollaran métodos que permitiesen firmar digitalmente pero que no pudiesen ser usados para encriptar. Así nació en 1991 el DSA para ser usado en el DSS [*Digital Signature Standard*; estándar de firma digital].

Sin embargo, se equivocaron, pues con una implementación general del DSA es posible encriptar (véase el capítulo 20 de Bruce Schneier [1996]).

Ahora bien, ¿la relajación de las restricciones a la exportación de criptografía significaron que no nos vigilaban más? No, en absoluto: el monitoreo pasó a ser mucho más inteligente e invisible para los no especialistas.[3]

En agosto de 1999 se desató una controversia cuando Andrew Fernandes denunció la existencia de una NSAKEY (una llave de la Agencia Nacional de Seguridad) en los sistemas Windows.[4]

Las técnicas más recientes se basan en sesgar la generación de números seudoaleatorios. Para mayor claridad recordemos que en el método RSA el primer paso es elegir dos primos distintos, p y q, que se multiplican para formar el módulo n, que va a ser parte de la clave privada y de la pública. Si un atacante puede factorizar n obtiene los primos y, a partir de allí, puede deducir la clave privada a partir de la pública, lo que supone el fin de la seguridad. Por lo tanto, imaginemos que nuestro generador de números seudoaleatorios solamente puede producir 10.000.000 de pares de primos distintos. Si tomamos n y lo dividimos por esos primos, encontraremos los factores rápidamente (o, mejor aún, calculemos el máximo común divisor de n con ellos y, cuando dé distinto de 1, significa que hallamos un factor).

Este sesgo de los generadores puede lograrse mediante ingeniosos métodos matemáticos o por errores de programación. La primera opción se debe, naturalmente, a objetivos de espionaje y se ocultan lo más posible. De la segunda

3 Véanse, por ejemplo, "Mentiras arriesgadas y sobre los gobiernos que se dejan seducir por ellas", en <www.hispalinux.es> y Maloof (2013).

4 Véase "_NSAKEY", en <www.en.wikipedia.org>.

posibilidad existen diversos ejemplos, como lo que encontró Luciano Bello en Debian, una distribución muy popular de Linux,[5] que evidencia una vez más que tener código abierto no es una garantía automática de seguridad.

Varios ejemplos de la primera opción han sido expuestos públicamente. Un caso resonante ocurrió en el año 2007 luego de que el NIST publicara un estándar para la generación de números seudoaleatorios (*Special Publication 800-90A*), que se cree firmemente que incluía una puerta trasera diseñada por la NSA relacionada con un método de encriptación mediante curvas elípticas llamado DUAL_EC_DRBG. Los investigadores Daniel Brown y Kristian Gjøsteen presentaron en la conferencia Crypto 2007 una debilidad esencial: algunas constantes del método tienen una relación matemática con un conjunto de números secretos y cualquiera que los conozca puede predecir con exactitud cuáles serán los números generados por el algoritmo.

¿Cómo intervino la NSA en esto? Al ser la principal agencia de criptografía del gobierno estadounidense, siempre contribuyó a realizar los estándares que define el NIST y fue esa agencia la que propuso incluir ese algoritmo. Pero en este caso, y tal como puntualizó Bruce Schneier (2007), el método de las curvas elípticas no solo tiene la gran debilidad mencionada. También es mucho más lento que las alternativas conocidas y las implicancias de esto son obvias.

Además, un muy reconocido investigador de la Escuela Politécnica de Lausanne, Arjen Lenstra, y su equipo estudiaron una gran cantidad de certificados digitales y encontraron que con una frecuencia inusual el máximo común divisor de sus módulos era distinto de 1, lo que indicaba que tenían un primo en común, eso se originaba en un generador defectuoso de números seudoaleatorios.[6] Afortunadamente el problema se restringía a dispositivos como *routers*, cortafuegos, etc., y no a sitios de comercio electrónico.

5. Véase "Luciano Bello: explotación del *bug*", en <www.youtube.com>.

Una posibilidad muy atractiva es utilizar el hardware de una computadora para generar números seudoaleatorios, por ejemplo midiendo variaciones de temperatura del procesador. Esta es una variable física que depende de lo que esté haciendo: si está realizando cálculos intensivos, la temperatura aumenta. Esta posibilidad existe en los procesadores de Intel a partir de la generación Ivy Bridge, pero se considera que hubo un acuerdo con la NSA para volver a afectar a la generación de números seudoaleatorios. Por cierto, Linus Torvalds, el creador de Linux, denunció este hecho.[7]

Lo que sí se hace actualmente es utilizar la opción RdRand que provee Intel tanto en Linux como en FreeBSD como una de las fuentes de aleatoriedad, agregando otras para no utilizarla en forma "pura". De hecho, en criptografía es muy frecuente usar varias fuentes de bits aleatorios y mezclarlas mediante operaciones booleanas como el XOR que vimos al tratar el anotador de única vez.

En conclusión, podemos decir que los métodos criptográficos pueden ser inviolables siempre y cuando se eviten cuidadosamente los talones de Aquiles originados en generadores seudoaleatorios que no hayan sido auditados con seriedad.

6. Véase, por ejemplo, Leyden (2012).

7. Véase Clarke (2013).

ESTEGANOGRAFÍA

La esteganografía es el arte de ocultar mensajes dentro de otros mensajes aparentemente inocuos. Sus orígenes se remontan a tiempos muy remotos. Por ejemplo, en la época de Heródoto de Halicarnaso se fabricaban cuadernos con tablitas de madera cubiertas con cera para poder escribir sobre ellas. Aparentemente en su obra de nueve tomos, *Historia*, describe que una persona removió la cera de una tablilla, grabó un mensaje directamente sobre la madera y volvió a cubrirla con cera. También menciona que a un esclavo se le rapó parte del cabello, se le tatuó un mensaje en el cuero cabelludo y, una vez que el cabello volvió a crecerle, se lo mandó como mensajero con la instrucción explícita de que lo volvieran a rasurar.

La esteganografía tuvo un período muy importante durante la Segunda Guerra Mundial con el uso de microfilms con los que se podía reducir una fotografía de una hoja de texto a un punto que luego se colocaba sobre una "i" o como signo de puntuación. Para leerlo bastaba un microscopio.

Los mensajes pueden ocultarse en audio digital, que es la codificación digital mediante números enteros de una señal eléctrica que representa una onda sonora, dado que esos números pueden asociarse a símbolos de un alfabeto.

Otra opción usada por los responsables de los atentados del 11 de septiembre de 2001 fue ocultar sus mensajes en imágenes. Por ejemplo, una fotografía en *true color* (color verdadero) caracteriza cada pixel por 24 bits. Recordemos que un pixel es la menor unidad homogénea en color en una imagen, y que esta se forma por una sucesión de pixeles. Con 24 bits se pueden formar 16.777.216 combinaciones distintas de colores. Ahora bien, si por ejemplo establecemos

como clave que el tercer bit de cada pixel corresponde al mensaje para transmitir la letra D = 01000100, solo debemos cambiar los correspondientes bits de nuestra foto con un software adecuado. Y si originalmente había un 0 y lo transformamos en un 1, o viceversa, el color cambiará en forma prácticamente imperceptible para el ojo humano.

Lo anterior se puede sofisticar mucho más, por ejemplo transmitiendo sucesiones encriptadas de enteros entre 1 y 24 que indiquen qué bits tomar de una imagen.[8]

USO DE WI-FI Y CELULARES

El área de software y servicios móviles es tal vez la más dinámica e innovadora de las TICs. Los servicios bancarios, financieros y de pagos móviles, los servicios móviles para viajeros y los entornos para la creación y provisión de contenidos basados en los usuarios son solamente algunos ejemplos bien conocidos de las nuevas direcciones en las aplicaciones de las tecnologías de la información. En efecto, los "usuarios comunes" se vuelven crecientemente creadores y proveedores de contenidos; por cierto, generarlos con las cámaras de sus teléfonos móviles es solo el comienzo. En este contexto, el tratamiento de la seguridad y la confiabilidad presenta desafíos tanto en materia de políticas y regulaciones como tecnológicos.

8. Es interesante consultar Camacho (2013) y descargar programas como Camouflage u OpenPuff. En Ortega (2010) figuran varios programas esteganográficos para ocultar información en archivos de audio, imágenes, archivos de texto o PDF.

El primer punto que debe considerarse es que los celulares son radios y, como tales, se pueden escuchar, interferir, etc., y lo mismo sucede con las redes inalámbricas.

Esto dio lugar al desarrollo de métodos de encriptación de las comunicaciones como WEP [*Wired Equivalent Privacy*; privacidad equivalente a cableado], el sistema de cifrado incluido en el estándar IEEE.802.11 como protocolo para redes Wi-Fi que permite encriptar la información que se transmite.

Desde sus comienzos, en 2001, diversos criptógrafos identificaron varias debilidades serias. En consecuencia, hoy en día una protección WEP puede ser violada en pocos minutos con softwares a los que se puede acceder fácilmente.[9]

La mejor opción es utilizar WPA2 [*Wi-Fi Protected Access 2*; acceso protegido Wi-Fi 2], que emplea el método de encriptación AES.

CONSEJOS PRÁCTICOS PARA AUMENTAR LA SEGURIDAD DE UNA RED WI-FI

- Establecer un nombre para su red Wi-Fi que no sea genérico como el que viene de fábrica con el *router* (hay herramientas de *hacking* que buscan los nombres habituales de las redes). Para ello, escribir en el navegador que utilice 192.168.1.1 o 192.168.0.1 y luego introducir el nombre de su usuario y la clave de acceso.
- Definir el método de encriptación como WPA2.

9. Véase, por ejemplo, "Crackear WEP usando wifislax 3.1 y airoscript", en <www.youtube.com>.

- Definir una clave fuerte (10 o más caracteres) sustituyendo la que viene de fábrica, que es normalmente "admin". Al respecto, es recomendable usar un generador de contraseñas como el que mencionamos en el apartado "Manejo de contraseñas" del capítulo 3.
- Una posibilidad muy recomendable es determinar la dirección MAC de cada dispositivo y habilitarlos a usar la red según esta. Para ello, en Windows haga lo siguiente:
  - vaya a Inicio;
  - haga clic en Ejecutar;
  - escriba cmd y haga clic en Aceptar;
  - en la ventana que se abrió escriba ipconfig/all y presione Enter;
  - la última línea contiene la dirección física (es la MAC) que debe ser listada en la pantalla de configuración del *router* inalámbrico para habilitar su conexión a la red.

Hay que tener en cuenta que cada vez que se agregue un dispositivo a la red, como una nueva PC, *notebook*, tableta o teléfono inteligente, hay que incorporarlo a la lista de los autorizados a usar la red. Esto impide que otros se conecten, aun cuando hayan quebrado la clave de acceso.

CONSEJOS PRÁCTICOS PARA AUMENTAR LA SEGURIDAD DE LOS CELULARES

Un teléfono inteligente es una pequeña computadora, menos vulnerable que los equipos que utilizan Windows, pero también sujeta a ataques de distinta índole. Por eso, es conveniente:

- No almacenar datos importantes, como números de tarjetas de crédito, claves o información personal (domicilios), etc.
- Proteger el teléfono con una clave de modo tal que cuando se apague sea imprescindible usarla para desbloquearlo.
- No instalar aplicaciones de origen no comprobable.
- Desactivar el Bluetooth.
- Instalar un antivirus, sobre todo si el sistema operativo es Android y tiene localizador del equipo que permita ubicarlo en caso de robo y bloquearlo en forma remota.[10] Estos programas también informan si se le ha cambiado la tarjeta SIM [*Subscriber Identity Module*; módulo de identificación de abonado], que guarda la clave de suscripción del usuario para habilitar su conexión a la red móvil.
- Registrar el número IMEI [*International Mobile Equipment Identity*; identidad internacional de equipo móvil]. Este se encuentra debajo de la batería y consta de quince dígitos (también se lo puede obtener escribiendo "*#06#").

  El IMEI se debe reportar a la compañía proveedora del servicio para que bloquee el equipo. Esto es así porque, cuando encendemos el teléfono, la operadora lo identifica por el IMEI y puede consultar en una base de datos llamada EIR [*Equipment Identity Register*; registro de identidad de equipo], en la cual hay tres listas diferenciadas por colores:

10. Véase, por ejemplo, "Rastrear un celular o un *laptop*", en <www.windowsespanol.about.com>.

  - la blanca, con los teléfonos sin problemas;
  - la gris, con los teléfonos habilitados para hacer y recibir llamadas, pero que pueden ser monitoreados para descubrir quién lo está usando mediante la información grabada en la tarjeta SIM, y
  - la negra, que contiene la identificación de los teléfonos robados o usados en forma ilegal que se bloquean, o sea que no pueden recibir ni efectuar llamadas. Todas las empresas proveedoras consultan esa base de datos e impiden que un teléfono que se encuentre en la lista negra pueda ser usado.

- Todo teléfono inteligente viene con un programa para conectarlo a una computadora que permite hacer una copia de seguridad de toda la información [*backup*].

BORRADO INSEGURO DE ARCHIVOS

Como es sabido, el simple borrado de un archivo no elimina la información y existen muchas maneras de recuperarla.

El "borrado seguro" [*wiping*] sobrescribe el área ocupada por un archivo con ceros, con unos o con caracteres aleatorios, y cada vez más encontramos en las pericias que esta técnica ha sido utilizada aun por personal no especializado.

Dado que el simple sobregrabado de ceros no brinda demasiada seguridad en el borrado de información confidencial, hay técnicas mucho más sofisticadas, como el estándar del Departamento de Defensa de Estados Unidos (DoD 5220.22-M), que usa tres ciclos de sobregrabación (ceros, unos y bits al azar) y uno final de lectura para verificación.

Hay que tener en cuenta que se trata de una operación lenta que puede llevar horas con un disco rígido de la capacidad actual, por lo que no puede ser empleada cuando el tiempo apremia. Hay otros estándares aún más exigentes, como el Gutmann, que hace treinta y cinco ciclos de sobregrabación de ceros y unos aleatorios (Gutmann, 1996).

Pero ¿qué sucede al grabar y sobregrabar? Cuando se escriben los datos, las cabezas del disco definen la polaridad de la mayoría de los dominios magnéticos, pero no de todos. El efecto de sobrescribir un 0 con un 1 da aproximadamente 0,95, y el de sobrescribir un 1 con otro 1 da aproximadamente 1,05. Los equipos de lectura normales interpretan ambos como un 1, pero con equipos muy especializados es posible descubrir la historia de la sobregrabación.

Asimismo, existen técnicas modernas de nanotecnología, como SPM [*Scanning Probe Microscopy*; microscopio por sonda de barrido], y su continuador, MFM [*Magnetic Force Microscopy*; microscopio de fuerza magnética], que permiten conseguir imágenes derivadas de la magnetización remanente de los bordes de las pistas de los discos, que forman una "fotografía" de la historia de grabación.

También se usan aparatos llamados *degaussers* (desmagnetizadores) para borrar información mediante la creación de fuertes campos magnéticos.

Por eso, como dice J. J. Sawyer (s. f.), un conocido experto de los Estados Unidos:

> Hasta hace poco tiempo se creía que ejecutar el *wiping* reiteradamente sobre una misma área del disco eliminaba todas las posibilidades de recuperación, y por eso el Departamento de Defensa de los Estados Unidos había generado el estándar DoD (directiva DoD 5220.22-M-Sup 1 capítulo 8, "Degaussing is More Reliable than Overwriting Magnetic Media" [DoD, 2005: 18]), recomendando repetir el proceso siete veces (y otros científicos llegaron a promover treinta y tres "pasadas").

Hoy en día para proteger los archivos *Top Secret* la única recomendación es destruir el disco.

En el mundo existen diversos laboratorios para efectuar los análisis forenses de medios magnéticos que, en general, pertenecen a gobiernos, aunque hay algunos privados.

USO DEL CORREO ELECTRÓNICO

La existencia de Internet, así como el uso masivo del correo electrónico, han ofrecido innumerables beneficios a la humanidad, pero traen aparejados diversos problemas que deben ser tenidos en cuenta.

Desafortunadamente, el correo electrónico es un medio propicio para que personas inescrupulosas intenten atentar contra la seguridad de los usuarios. La distribución masiva de virus, *malware*, etc., mediante archivos infectados es un tema cotidiano. También permite el *spam* (correos no solicitados), cuyo nombre se debe a un tipo de carne enlatada que los soldados estadounidenses recibían durante la Segunda Guerra Mundial, sea para realizar acciones de marketing como para otros fines, así como el *phishing*, del que hablaremos luego.

El *scam* (estafa) consiste en mandar correos anunciando que se han ganado premios falsos, proponiendo negocios fabulosos, o métodos para recibir millones de parte de fun-

cionarios corruptos que están en países con guerras internas, u ofreciendo trabajo, etc. Su objetivo obvio es sacarles dinero a los que responden.

CONSEJOS PRÁCTICOS PARA EL USO DEL CORREO ELECTRÓNICO

- Cuando mande *e-mails* a varios destinatarios use el campo CCO o BCC (copias ocultas) para no exponer las direcciones de sus contactos, pues estas son recolectadas por los *spammers* para sus propios fines o para venderlas por la Web.
- En la medida de lo posible evite publicar su correo en las redes sociales, blogs, etc. La mayoría de las listas de discusión por correo electrónico crean archivos accesibles vía Web. Controle si la lista que le interesa archiva los mensajes y, si lo hace, si estos son públicos o es necesario suscribirse para acceder a ellos.
- Los *hoax* (bulos) consisten en difundir masivamente noticias impactantes aunque no sean reales mediante cadenas de correos en las que se pide que uno a su vez las retransmita. Uno de sus objetivos es capturar direcciones de *e-mail* para los *spammers*.
- Es útil disponer de cuentas alternativas para registrarse en sitios dudosos con las direcciones que usa para temas privados.
- Nunca conteste o abra un correo de origen desconocido, mucho menos si contiene archivos adjuntos.
- Impida la visualización de imágenes en el cuerpo de los mensajes de sus correos. Cualquier programa de

webmail actual cuenta con esa función. Solamente descargue o visualice imágenes cuando tengan un origen confiable.
- No acceda a los sitios web a los que reenvía un correo de origen desconocido.
- No anule la suscripción a boletines de noticias [*newsletters*] a los que nunca se haya suscripto. Una técnica común usada por correos electrónicos no deseados es enviar miles de boletines de noticias falsos de organizaciones con un enlace del tipo "Darse de baja" o "*Unsubscribe*". Los usuarios de correo que introducen su dirección de correo electrónico en ese *link* (enlace) de la supuesta lista en muchos casos serán víctimas de *spam* o *phishing*. Si no recordamos habernos suscrito a un boletín de noticias, es mejor marcar la dirección de correo electrónico como *spam* para que sus *e-mails* sean filtrados por el programa de correo.
- Evite enviar datos de pagos o referentes a cuentas bancarias aunque utilice programas de correo como Hotmail, Google Chrome, etc., que se conectan mediante https.
- Borre periódicamente el historial de navegación y las *cookies* (por ejemplo con CCleaner). Esto es especialmente importante si consultó su correo desde un sitio público.
- Siempre cierre su cuenta de correo antes de cerrar el navegador.
- Si desea firmar y encriptar sus correos, utilice certificados digitales. Otra posibilidad es encriptar archivos y adjuntarlos a sus correos, siempre y cuando los destinatarios posean las claves correspondientes.

PHISHING

Un intento de estafa *phishing* (término que se pronuncia igual que el gerundio "*fishing*" –pescando– y significa pesca de datos) es un tipo de fraude por correo electrónico que trata de engañar a los destinatarios para que revelen información personal (como sus números de tarjetas de crédito o débito, información sobre sus cuentas de cheques, números de seguro social o contraseñas de cuentas bancarias) a través de sitios web simulados.

Generalmente imitan páginas que los destinatarios conocen, como las de entidades bancarias, para engañarlos. Aunque a veces estos intentos de cometer fraudes son bastante sofisticados, existen algunas características visibles:

- Piden información personal con el pretexto de que la casilla de correo está llena o que es necesario actualizar los datos. En este caso debe recordar que los bancos jamás solicitan información confidencial por correo.
- Provienen, por ejemplo, de compañías que uno no conoce o el *e-mail* que se recibe tiene errores en el nombre del destinatario o usan fórmulas genéricas, como "estimado cliente", sin dar ningún nombre.
- Avisan que se ha cometido un fraude con la tarjeta número xxxxxxxxxx2304; estos últimos dígitos coinciden con los de una tarjeta verdadera y generalmente roban la información de sitios que exhiben los cuatro números finales. El usuario puede caer en la trampa completando los datos faltantes.
- A menudo incluyen amenazas, por ejemplo, "informan" que si no se suministra la información solicitada en un plazo perentorio, la cuenta será bloqueada, etc.

- En los casos de los bancos que utilizan tarjetas de coordenadas para completar transferencias, los correos de *phishing* solicitan todas las coordenadas y desaparece la barrera para realizar transferencias. En este caso también debe recordar que los bancos nunca piden esa información.
- Es frecuente que estos correos incluyan *links* a páginas que simulan ser las verdaderas pero que no lo son. Nunca acceda a través de ellas.
- A veces estos correos solicitan donaciones para las víctimas de catástrofes en alguna parte del mundo y, para hacerlas, piden los datos de las tarjetas.

Cuando escribimos en un navegador un nombre de un sitio al que queremos acceder, este debe ser transformado en la dirección IP correspondiente. Esto se realiza a través del llamado DNS [*Domain Name Server*; servidor de nombre de dominio]. Un DNS utiliza bases de datos distribuidas para tratar de resolver lo que el usuario desea de modo eficiente. Por ejemplo, como es frecuente que requiera sitios de su país, que se encuentran en una base de datos local, esos son los primeros que aparecen y, en caso de que no existan localmente, procede a buscar en la estructura jerárquica internacional.

Una de las variantes más peligrosas del *phishing* es el llamado *pharming*, que consiste en modificar el DNS para guiar al usuario a una página web falsa. Esto lo conseguían mediante *malware* que modificaba el DNS local residente en un archivo llamado Hosts, que era el primero que se consultaba ante una solicitud de dirección. Pero el crecimiento de la Web modificó este esquema inicial y, por ello, los ataques actuales son más complejos.

Hasta ahora mencionamos los correos electrónicos como fuente principal del problema. Pero también existen variantes como el *smishing*, realizado a través de teléfonos celulares, o el *vishing*, mediante telefonía IP.

Los intentos de *phishing* vienen creciendo sistemáticamente, tal como se reportó en España en 2014.

Un estudio reciente elaborado por los Servicios de Inteligencia para la Seguridad de IBM asegura que los ataques de estas características habían crecido un 5000% durante el año 2013.

En la Web se pueden encontrar numerosos informes de casos de *phishing* realizados prácticamente en todo el mundo.

¿Hacia una seguridad inquebrantable?

Acerca de la computación y la criptografía cuántica

CRIPTOGRAFÍA CUÁNTICA

La criptografía cuántica se basa en los principios de la mecánica cuántica. Una de las propiedades esenciales que la caracterizan se origina en el principio de incertidumbre que Heisenberg enunció en 1925. Se trata de una diferencia esencial entre la física newtoniana y la física cuántica que, en términos simples, afirma que el propio proceso de observar cambia lo que se está observando a nivel de las partículas subatómicas. La paradoja consiste en que no se pueden medir simultáneamente dos propiedades relacionadas de una partícula, como su posición exacta y su momento (producto de la velocidad por la masa). Por ejemplo, cuando un fotón emitido por una fuente luminosa impacta con un electrón ubica su posición pero, al mismo tiempo, cambia su velocidad.

Una analogía podría ser la siguiente: supongamos que hay un gato en una habitación a oscuras y queremos determinar su posición. Entramos a la habitación, prendemos una luz y el gato salta. Nunca podremos saber dónde estaba originalmente.

Aquí podemos citar a Niels Bohr, uno de los padres de esta rama de la física: "Cualquiera que pueda contemplar la mecánica cuántica sin sentirse mareado es porque no la ha entendido". Y realmente existen muchos fenómenos, verificables en la práctica, que no se entienden en absoluto.

Como dice Juan Pedro Hecht (2012) en su excelente introducción a estos temas para los lectores con cierta formación científica: "En general el mundo clásico es continuo y causal; el cuántico, discreto y aleatorio".

El primer protocolo de criptografía cuántica llamado BB84 fue publicado en 1984 por Charles Bennett (un químico doctorado en Harvard e investigador del T. J. Watson Research Center de IBM) y Gilles Brassard (doctor en Computación por la Universidad de Cornell), quienes se inspiraron en la idea –nada práctica– de Stephen Wiesner de crear "dinero cuántico".[1]

El principio esencial es que, cuando dos entidades se comunican para establecer una clave, cualquier espía que la intercepte necesariamente va a alterar la comunicación. Por lo tanto, los participantes se darán cuenta y ninguna información podrá filtrarse. Esto se logra mediante fotones polarizados.

La polarización es una propiedad de la luz cuyo origen cuántico indica que el campo electromagnético posee dos componentes independientes. Un solo fotón tiene una polarización que está definida por un ángulo. En un laboratorio de óptica cuántica se puede generar un fotón con una polarización dada, y también se dispone de detectores para medir la polarización horizontal, vertical o en cualquier otro

1. Véase, por ejemplo, Brassard (2005).

ángulo. Si un fotón en polarización horizontal incide en un detector ajustado a polarización vertical, la probabilidad de descubrirlo será cero, si incide uno con polarización vertical será del 100% y si incide uno a 45 grados será del 50%.

Supongamos que usamos fotones con cuatro polarizaciones posibles, algo que podemos hacer como mencionamos en el párrafo anterior, y los representaremos mediante:

Detector	0	1
+	↑	→
x	↗	↘

El proceso entre dos entidades A y B para arribar a una clave no detectable es el siguiente:

- La entidad A elige una sucesión de bits al azar, como 1 0 1 0 0 0 1, y la transmite a través de fotones polarizados como →↗↘↑↑↗→ mediante fibra óptica. (Nótese que, por ejemplo, el primer 1 se transmitió como → pero el segundo lo hizo como ↘.)
- La entidad B no sabe qué detectores se usaron para definir la secuencia transmitida, así que elige uno u otro al azar, los anota, y lo mismo hace con los resultados obtenidos.
- A y B se comunican mediante un canal público informando a la otra parte la sucesión de detectores utilizados pero sin especificar en cada caso cuál de las dos posibilidades fue la elegida.
- A y B descartan los bits correspondientes a los casos donde los detectores no coincidían. Por lo tanto, en

promedio se quedarán con la mitad de los bits de la secuencia original.

En el ejemplo anterior la sucesión de detectores usados por A era + x x + + x +.

El primer símbolo transmitido era → ; y si B elige como primer detector x ese símbolo se interpretará como ↗ o ↘ no pudiendo de ese modo llegar a una conclusión acerca de si era un 0 o un 1, pero si eligió el correcto, entonces no hay ninguna duda acerca del resultado.

En principio un espía estaría en la misma situación que B al no saber qué detector usar. Pero si intenta medir los fotones usando detectores al azar, por el principio de incertidumbre de Heisenberg inevitablemente va a introducir una perturbación cuando elija uno equivocado. Entonces, para completar el protocolo, A y B pueden explicitar algunos bits de la sucesión final para ver si coinciden o calcular un *hashing* de la solución correcta.

Por ejemplo, supongamos que de la sucesión original 1 0 1 0 0 0 1 solamente queda 1 0 0 0 1, su *hashing* SHA-1 es 6C447A8FE7677DDC4C4CD2EFDDCFE650E4E6C706, resultado que ambos pueden calcular e intercambiar para controlar que coincidan. Si no lo hacen, significa que hubo interferencias y el proceso se descarta.

Naturalmente en una aplicación práctica la sucesión tiene gran longitud y puede usarse tanto para el anotador de única vez, como clave de un método simétrico o como semilla de un generador aleatorio de claves.

Una regla aceptada es que si la tasa de bits que se van a descartar no excede el 20%, el intercambio de claves se considera seguro. Sin embargo, un grupo de físicos de la

universidad de Toronto logró quebrar un sistema cuántico comercializado por la compañía suiza ID Quantique (Feihu *et al.*, 2010).

En la actualidad la compañía norteamericana MagiQ Technologies y la suiza ID Quantique son las únicas que venden equipos de distribución de claves mediante fotones; cuestan unos 100.000 euros y, en el caso de los distribuidos por la empresa MagiQ Technologies, solo funcionan con total seguridad en enlaces dedicados[2] de fibra óptica inferiores a veinticinco kilómetros.

En octubre de 2013 se completó la instalación en los Estados Unidos del primer centro de distribución de claves cuánticas utilizando equipos de ID Quantique. Un problema es que la distancia máxima para transmitir es de un poco más de 96 kilómetros, con lo que es necesario utilizar equipos de repetición, supuestamente confiables, para alcanzar los sitios que deben conectarse.

COMPUTADORAS CUÁNTICAS

Como los aspectos técnicos de las computadoras cuánticas exceden los alcances del presente libro, solo nos vamos a enfocar en sus posibles implicancias para la seguridad informática.

Las computadoras cuánticas en vez de usar bits como las máquinas convencionales emplean qubits que almacenan los valores 0 y 1 simultáneamente, lo que les permitirá ser

2. Un enlace dedicado es una conexión física entre dos puntos (por ejemplo, la casa central de un banco y una de sus sucursales) que no se comparte con terceros.

mucho más rápidas. Es cierto que todavía quedan muchos problemas tecnológicos por resolver, pero lo más probable es que se logre hacerlo en un futuro no muy lejano.

Dado que el paradigma computacional es muy distinto, se crearon algoritmos especiales como el de Schor (1997), el cual en teoría permitirá factorizar enteros de gran longitud en tiempos razonables si se llega a disponer de una máquina cuántica con una gran cantidad de qubits.

La racionalidad de utilizar mecanismos cuánticos se basa en la creencia de que en pocos años la disponibilidad de computadoras cuánticas permitirá quebrar los métodos más tradicionales. Se escuchan frases como "Muchos criptosistemas, como el RSA, se tornarán obsoletos si el algoritmo de Schor se llega a implementar en una computadora cuántica real". Pero la situación actual es que en el año 2012 se logró el "récord" de factorizar el número 143, lo que muestra que todavía queda mucho camino por recorrer (Zyga, 2012). Y, de hecho, varios científicos dudan de que ese algoritmo vaya a tener alguna vez implicaciones prácticas.

Pero, independientemente de las opiniones contrapuestas, en 2014 se supo, por las revelaciones de Edward Snowden, que la NSA está trabajando activamente en el desarrollo de computadoras cuánticas para quebrar códigos como parte del proyecto *Penetrating Hard Targets* (penetrando objetivos difíciles), que cuenta un presupuesto de unos 80 millones de dólares.

¿Cómo amenazan nuestra privacidad?

Acerca de la seguridad, privacidad, confiabilidad, los delitos informáticos y la información pública

INCIDENTES EN EL MUNDO REAL

La siguiente es una muy breve lista de problemas reales de seguridad:

- *The Washington Post* publicó el 6 de agosto de 2014 que unos *hackers* rusos robaron más de un billón de *passwords*.
- Muchas listas de sitios *hackeados* se encuentran en la Web.
- Nuevos casos de *phishing* ponen en peligro las cuentas de los usuarios de varios bancos españoles.
- La Asociación de Internautas (AI) ha informado que se están produciendo envíos masivos de correos electrónicos fraudulentos que solicitan datos personales y claves de acceso a los servicios de los bancos BBVA, Banesto, Cajamar y Banco de Valencia.
- Anonymous es una agrupación de personas dedicadas

a *hackear* sitios de gobiernos, empresas, etc., que apoyan la censura en la Web.[1]

- Facebook es un objetivo frecuente de ataques. En el año 2012 fue atacado por el gusano Ramnit, que robó unos 45.000 *passwords*. Esta empresa y Google han sufrido el robo de millones de *passwords* (véase por ejemplo Bogursky, 2013).
- Twitter también es un objetivo deseable. En el año 2013 le robaron los datos y *passwords* de 250.000 usuarios.
- El 26 de febrero de 2014 se anunció que 360 millones de *passwords* estaban en venta en sitios de delincuentes (Goodin, 2014).
- El 31 de enero de 2014 Yahoo anunció que había sido víctima de un ataque coordinado en el que consiguieron robarles las claves de *e-mail* de muchos usuarios.
- Hubo muchos casos de *hackeo* de cuentas de Hotmail. Uno que se hizo famoso fue cuando *e-mails* supuestamente enviados por el secretario de Justicia inglés Jack Straw llegaron a cientos de usuarios con el objeto de cometer fraudes. En 2014 diversos usuarios reportan incidentes similares.

Un estudio de la empresa Symantec (2014) clasificó los problemas de seguridad del año 2013 del siguiente modo:

1. Una lista de los sitios que *hackearon* se encuentra en Raza (2012).

Cuadro 7.1

Hacking	35%
Divulgación accidental de claves y/o documentos	29%
Robo o pérdida de un ordenador	27%
Apropiación de claves y/o documentos por parte de personal interno	6%
Problemas desconocidos	2%
Fraudes	2%

En el año 2013 se detectaron 6787 nuevas vulnerabilidades (en el año 2012 hubo 5291, lo que muestra un fuerte incremento). Muchas de ellas fueron aprovechadas mediante los llamados "ataques de día cero", que consisten en diseñar un código malicioso para efectuar ataques mientras las vulnerabilidades no son conocidas ni por los usuarios ni por los fabricantes de los productos de software. En los Estados Unidos existe una base de datos pública de vulnerabilidades.[2]

Por estos motivos lo recomendable es tener todo el software actualizado pues las empresas proveedoras van corrigiendo los productos en función de las vulnerabilidades detectadas.

También es interesante ver los datos acerca del porcentaje de correos electrónicos que contenían enlaces a sitios web dañinos. El valor máximo durante el año 2013 se dio en abril y fue de aproximadamente el 42%.

2. Véase <nvd.nist.gov>.

SEGURIDAD, PRIVACIDAD, CONFIABILIDAD Y DELITOS INFORMÁTICOS

Las redes digitales, principalmente Internet, constituyen hoy en día una parte esencial de nuestras vidas. Los datos privados son cada vez más vulnerables y los problemas relacionados con la seguridad y la confiabilidad se tornan un factor clave para los ciudadanos que se conectan a sistemas cuya complejidad excede los conocimientos técnicos que poseen. Este problema es particularmente serio en el caso del uso de celulares.

Virus, *spyware*, *phishing*, etc., son delitos cada vez más sofisticados, pero la gente se ve obligada a confiar en ambientes que no puede juzgar adecuadamente.

Este marco general conspira contra la construcción de una sociedad de la información que pueda generar desarrollo, prosperidad y equidad social. Para que así sea se necesita utilizar medios de comunicación de muy bajo costo, como Internet, pero que son intrínsecamente inseguros. En contrapartida, es necesario adaptar las TICs a las necesidades de la economía y la sociedad, y asegurar que se transformen en herramientas útiles para la innovación en esas áreas. El punto de partida para lograrlo es fomentar la confianza y salvaguardar la seguridad en un mundo cada vez más interconectado por redes.

TRES TÉCNICAS DE CIBERTECNOLOGÍA QUE AMENAZAN LA PRIVACIDAD

1. *Data gathering* (recolección de datos): se usa para acumular y registrar información personal, muy a menudo sin el consentimiento de los usuarios.

2. *Data exchanging* (intercambio de datos): se utiliza para transferir e intercambiar datos personales a través de bases de datos y entre ellas, también típicamente sin el consentimiento de los usuarios.
3. *Data mining* (minería de datos): sirve para buscar patrones en grandes bases de datos a fin de generar perfiles de consumidores basándose en los comportamientos descubiertos a partir de ciertos grupos.
   La minería de datos implica la recolección indirecta de información. A diferencia de la información personal, que está explícitamente almacenada en las bases de datos, la obtenida mediante esta técnica se obtiene a través del análisis de patrones implícitos –reales o ficticios– en los datos.
   Las actividades de minería de datos pueden generar clasificaciones o categorías nuevas incluso, a veces, no obvias. Los individuos, cuyos datos se han extraído mediante este proceso podrían identificarse, relacionarse o asociarse con ciertos grupos que ni siquiera saben que existen. Pueden sugerir, por ejemplo, que la persona pertenece a una nueva categoría que se acaba de "descubrir".
   Las leyes de privacidad no ofrecen protección respecto al uso que se hace de lo que se obtiene con este procedimiento. Dado que se pueden tomar decisiones importantes sobre esos individuos basándose en los patrones hallados, algunos de los usos de la minería de datos son especialmente preocupantes en lo que respecta a la privacidad personal. Se utilizan por ejemplo para desarrollar sistemas de *scoring*, que son los que habilitan o no a una persona para conseguir una tarjeta de crédito, una cuenta bancaria, o un préstamo.

DATAVEILLANCE

Los movimientos físicos de una persona mientras está comprando en una tienda son monitoreados por videocámaras.

Los automotores están vigilados en las autopistas mediante dispositivos como los *tags* de peaje.

Los coches, *laptops*, etc., tienen dispositivos de localización.

La cantidad de "clics" (teclas y clic del ratón) que se realizan al visitar un sitio web puede ser monitoreada y registrada.

LAS *COOKIES* DE INTERNET COMO UNA TÉCNICA DE VIGILANCIA/MONITOREO

La tecnología de las *cookies* permite a los propietarios de los sitios web reunir datos acerca de aquellos que acceden a sus sitios.

Permiten "capturar" las preferencias de navegación de cada uno cuando se encuentra en línea cada vez que se visita un sitio web y esa información puede recuperarse luego y ser remitida al sitio web la próxima vez que el usuario ingrese.

El intercambio de datos típicamente se realiza sin conocimiento ni consentimiento del usuario.

¿ES DEFENDIBLE EL USO DE *COOKIES*?

Muchos sitios web que utilizan *cookies* sostienen que están prestando un servicio a los usuarios que vuelven a sus sitios web dado que ajustan a su gusto los mecanismos de búsqueda de información.

También indican que gracias a ellas les pueden ofrecer una lista de preferencias para sus futuras visitas.

OPINIONES EN CONTRA DE LAS *COOKIES*

Algunos defensores de la privacidad indican que el monitoreo y registro de las actividades que los individuos realizan cuando están navegando es una violación a la privacidad.

También les preocupa que la información recolectada mediante ellas pueda llegar a venderse.[3]

PROBLEMAS CON LA INFORMACIÓN PÚBLICA

Considere dos escenarios posibles:

- Hacer las compras en un supermercado.
- Hacer las compras en un sitio en Internet.

Ambas revelan problemas de protección de la privacidad en público en un área de cibertecnología y minería de datos.

Los detalles de su compra en Internet le pertenece al dueño del sitio aunque esa información sea sobre usted.

¿Es correcto que el dueño del sitio posea estos datos y haga con ellos lo que quiera por el tiempo que quiera? ¿Es lícito que pueda venderlos, compartirlos o canjearlos?

3. Algunas *cookies* pueden encontrarse en: <www.cookiecentral.com>.

Es por estas razones, entre otras, que diversos gobiernos impulsan la necesidad de acordar un marco regulatorio internacional para el uso de Internet.

Bibliografía

Bennett, Charles y Brassard, Gilles (1984): "Quantum Cryptography: Public Key Distribution and Coin Tossing", ponencia presentada en la International Conference on Computers, Systems and Signal Processing, Bangalore, India, 10-12 de diciembre. Disponible en línea: <www.cs.ucsb.edu>.

Bogursky, Sasha (2013): "How to Protect Yourself: 2 Million Facebook, Google Accounts Compromised", 5 de diciembre. Disponible en línea: <www.foxnews.com>.

Brassard, Gilles (2005): "Brief History of Quantum Cryptography: A Personal Perspective", Université de Montréal, Département d'informatique et de recherche opérationnelle. Disponible en línea: <www.arxiv.org>.

Camacho, Manuel (2013): "Esteganografía. El arte de camuflar archivos", 11 de abril. Disponible en línea: <www.hacking-etico.com>.

Clarke, Gavin (2013): "Torvalds Shoots Down Call to Yank 'Backdoored' Intel RdRand in Linux Crypto", 10 de septiembre. Disponible en línea: <www.theregister.co.uk>.

eMarketer (2013): "Ecommerce Sales Topped $1 Trillion for First Time in 2012", 5 de febrero. Disponible en línea: <www.emarketer.com>.

Feihu, Xu; Bing, Qi y Hoi-Kwong, Lo (2010): "Experimental Demonstration of Phase-Remapping Attack in a Practical Quantum Key Distribution System", *New J. Phys.*, vol. 12, noviembre. Disponible en línea: <www.iopscience.iop.org>.

Garlinski, Józef (1979): *The Enigma War*, Nueva York, Scribner.

Goodin, Dan (2014): "360 Million Recently Compromised Passwords for Sale Online", 26 de febrero. Disponible en línea: <www.arstechnica.com>.

Gutmann, Peter (1996): "Secure Deletion of Data from Magnetic and Solid-State Memory", Department of Computer Science, University of Auckland. Disponible en línea: <www.cs.auckland.ac.nz>.

Halsall, Paul (1998): "Suetonius (c.69-after 122 CE): *De Vita Caesarum, Divus Iulius* (The Lives of the Caesars, The Deified Julius), written c. 110 CE", *Internet Ancient History Sourcebook*, octubre. Disponible en línea: <www.fordham.edu>.

Hecht, Juan Pedro (2012): *Fundamentos de computación cuántica*, Madrid, Editorial Académica Española.

Kahn, David (1996): *The Code Breakers*, Nueva York, Scribner.

Leyden, John (2012): "Experts: RSA Weak Keys Flaw Restricted to Network Devices", 21 de febrero. Disponible en línea: <www.theregister.co.uk>.

Maloof, F. Michael (2013): "NSA Has Total Access via Microsoft Windows", 23 de junio. Disponible en línea: <www.wnd.com>.

Ortega, Alberto (2010): "Herramientas de esteganografía", 1 de diciembre. Disponible en línea: <www.securitybydefault.com>.

Pérez, Miguel (2013): "La era post-PC y el triunfo de la movilidad", 22 de abril. Disponible en línea: <www.blogthinkbig.com>.

Raza, Sheeraz (2012): "The List of Websites Hacked, Defaced & Taken Down By Anonymous", 14 de febrero. Disponible en línea: <www.valuewalk.com>.

Recorded Future (2014): "How Al-Qaeda Uses Encryption Post-Snowden (Part 1)", 8 de mayo. Disponible en línea: <www.recordedfuture.com>.

Reuvers, Paul y Weierud, Frode (2009): "Enigma Family Tree", *Crypto Museum*, 10 de septiembre. Disponible en línea: <www.cryptomuseum.com>.

Ribagorda Garnacho, Arturo (2010): "Historia de la criptografía y su desarrollo en Europa", Universidad Carlos III, Madrid, España. Disponible en línea: <www.criptored.ump.es>.

Sawyer, Joshua J. (s.f.): "Magnetic Data Recovery: The Hidden Threat", Greenville, East Carolina University. Disponible en línea: <www.infosecwriters.com>.

Schneier, Bruce (2007): "Did NSA Put a Secret Backdoor in New Encryption Standard?", *Schneier on Security*, 15 de noviembre. Disponible en línea: <www.schneier.com>.

— (1996): *Applied Cryptography*, Nueva York, Wiley (2da. ed.).

Shor, Peter (1997): "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer", *SIAM J. Comput.*, 26(5): 1484-1509. Disponible en línea: <www.arxiv.org>.

Singh, Simon (1999): *The Code Book*, Nueva York, Doubleday.

Smith, Richard (2002): *Authentication: From Passwords to Public Keys*, Boston, Addison-Wesley.

Stoll, Clifford (1991): *El gato y el ratón*, Buenos Aires, Planeta.

Symantec (2014): *Informe sobre las amenazas a la seguridad en Internet*, vol. 19. Disponible en línea: <www.symantec.com>.

Zyga, Lisa (2012): "143 is Largest Number yet to be Factored by a Quantum Algorithm", 11 de abril. Disponible en línea: <www.phys.org>.